AF

TRANSMITTAL OF APPEAL BRIEF (Large Entity)	Docket No. 200-0665

In Re Application Of: **Joseph G. Walacavage et al.**

OIPE JUL 19 2007 PATENT & TRADEMARK OFFICE

Application No.	Filing Date	Examiner	Customer No.	Group Art Unit	Confirmation No.
09/965,904	September 28, 2001	J. Proctor	33481	2123	4251

Invention: **METHOD OF LOGICAL MODELING OF OPERATOR INTERACTION WITH PROGRAMMABLE LOGIC CONTROLLER LOGICAL VERIFICATION SYSTEM**

COMMISSIONER FOR PATENTS:

Transmitted herewith is the Appeal Brief in this application, with respect to the Notice of Appeal filed on:

The fee for filing this Appeal Brief is: **$500.00**

☐ A check in the amount of the fee is enclosed.

☐ The Director has already been authorized to charge fees in this application to a Deposit Account.

☒ The Director is hereby authorized to charge any fees which may be required, or credit any overpayment to Deposit Account No. 06-1510. I have enclosed a duplicate copy of this sheet.

☐ Payment by credit card. Form PTO-2038 is attached.

WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.

Signature

Dated: July 16, 2007

Daniel H. Bliss (Reg. No. 32,398) [0693.00257]
Bliss McGlynn, P.C.
2075 West Big Beaver Road, Suite 600
Troy, Michigan 48084
(248) 649-6090

Record I.D. 81087874

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to "Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450" [37 CFR 1.8(a)] on July 16, 2007
(Date)

Signature of Person Mailing Correspondence

Daniel H. Bliss

Typed or Printed Name of Person Mailing Correspondence

cc:

P30LARGE/REV08

JUL 1 9 2007

IAP95

PATENT & TRADEMARK OFFICE

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Art Unit:	2123	)
Examiner:	J. Proctor	)
Applicant(s):	J. G. Walacavage et al.	)
Serial No.:	09/965,904	) **APPEAL BRIEF**
Filing Date:	September 28, 2001	)
For:	METHOD OF LOGICAL MODELING OF OPERATOR INTERACTION WITH PROGRAMMABLE LOGIC CONTROLLER LOGICAL VERIFICATION SYSTEM	)

Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450

Sir:

By Notice of Appeal filed May 14, 2007, Applicants have appealed the Final Rejection dated February 14, 2007 and submit this brief in support of that appeal.

REAL PARTY IN INTEREST

The real party in interest is the Assignee, Ford Global Technologies, Inc.

RELATED APPEALS AND INTERFERENCES

There are no related appeals or interferences regarding the present application.

CERTIFICATE OF MAILING: (37 C.F.R. 1.8) I hereby certify that this paper (along with any paper referred to as being attached or enclosed) is being deposited with the U.S. Postal Service with sufficient postage as First Class mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, Virginia 22313-1450 on July 16, 2007, by ______.
Daniel H. Bliss

07/20/2007 SSITHIB1 00000027 061510 09965904

01 FC:1402 500.00 DA

STATUS OF CLAIMS

Claims 1 through 15 have been rejected.

Claims 1 through 15 are being appealed.

STATUS OF AMENDMENTS

An Amendment Under 37 C.F.R. 1.116 was filed on April 16, 2007 in response to the Final Office Action dated February 14, 2007. An Advisory Action dated May 3, 2007 was issued and indicated that the Amendment under 37 C.F.R. 1.116 did not place the application in a condition for allowance. The Advisory Action stated that "Applicants may overcome the Schruben's reference description of randomly occurring events by amendments to the claim language that explicitly exclude any 'time dependence' for 'asynchronous operations'". Accordingly, a Second Amendment Under 37 C.F.R. 1.116 was filed on May 14, 2007 in response to the Final Office Action dated February 14, 2007 and Advisory Action dated May 3, 2007, that amended the independent claims to explicitly exclude any "time dependence" for "asynchronous operations". A second Advisory Action dated June 5, 2007 was issued and indicated that the Amendment under 37 C.F.R. 1.116 would be entered upon filing an appeal. A Notice of Appeal, along with the requisite fee, was filed on May 14, 2007. The Appeal Brief, along with the requisite fee, is submitted herewith.

SUMMARY OF THE CLAIMED SUBJECT MATTER

Independent Claim 1

The claimed subject matter of independent claim 1 is directed to a method of logical modeling operator interaction with a programmable logic controller logical verification system. [Referring to FIGS. 2 through 5, a method, according to the present invention, for logical modeling of operator interaction with the PLC logical verification system 16 of the system 10 is shown. In general, the user 12 models a human operator (not shown) in context of the part or product being developed, as a controller with a unique set of resources. In the present invention, a controller may be an operator, robot, PLC, or any programmable device. The resource assigned to the operator controller is a model of a physical manifestation of the operator in the workcell. For example, an operator resource for moving a part in a repetitive cycle might be an overhead crane that they directly control. Once the resource is bound to the operator controller, the PLC code or controller program can be written using conventional logic.] (FIGS. 2 through 5; Specification, page 7, line 15 through page 8, line 6).

The method includes the steps of constructing a flowchart that describes interaction of an operator in a workcell using a computer wherein such interaction comprises sequential operations and asynchronous operations, the asynchronous operations being not time dependent. [Referring to FIG. 2, the method begins in bubble 100 and advances to block 102. In block 102, the method includes selecting "commands" from a resource's capability to cause an action. The commands available to the operator flowchart (selected while the user 12 is in the action block) are determined by the collection of resources that have been bound to that controller. For example, a resource may be "Carry" for an operator to carry a workpiece from a

first location to a second location. The resource has at least one, preferably a plurality of capabilities. For example, a capability for the resource "carry" may be "lift" such that the operator lifts the workpiece before carrying the workpiece from the first location to the second location. It should also be appreciated that the method uses flowcharts to represent the cyclic logical behavior of the operator. It should further be appreciated that the purpose of the model is to verify the PLC logic by providing input signals to the PLC at desired times or based on conditional events.] (FIG. 2; Specification, page 9, lines 10 through 22 and page 9, lines 2 through 6).

The method also includes the steps of modeling the operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on predefined conditions described in the flowchart. [The method begins by writing a control model for an operator by the operator interaction design system 18. For example, the operator interaction design system 18 will create a control model definition that describes how an operator picks up a part, carries, and loads it into a fixture. As illustrated in FIG. 5, a flowchart of a method for a part flow in two stations or workcells is shown. The method begins in bubble 50 where an operator or pusher moves the part to Station 0 and on a transfer bar in block 52. The part moves forward in block 54, moves up in block 56, until the part is at a top in block 58. The part drops to the transfer bar in block 60. The part is at workspace 1 in block 62. The part moves forward to Station 1 in block 64. The part moves up in block 66, until the part is at the top in block 68. The part drops to a second transfer bar in block 70. The part is at workspace 2 in block 72. It should be appreciated that the control model is information that describes events, dependencies, and logical conditions. It should also be appreciated that the method uses flowcharts to represent the cyclic logical behavior of the operator. It should further be appreciated that the purpose of the model is to verify the PLC logic

by providing input signals to the PLC at desired times or based on conditional events. It should also be appreciated that the focus is on the logical representation of the operator and not the visual or spatial representations.] (FIGS. 2 and 5; Specification, page 8, lines 7 through page 9, line 9).

The method further includes the steps of testing the control model by a PLC logical verification system on the computer as to whether PLC logic for the workcell is correct. [Referring to FIG. 3, the method allows either the operator logic or PLC to test for status being returned from a resource through its input registers. The execution of the flowchart during simulation, which might be more than one based on different starting conditions, then proceeds by successive "command, test status" loops. The method begins with initializing the test in bubble 200. From bubble 200, the method advances to block 202 and idles the operator. For example, in block 202, the operator is set to idle where no work or motion is being performed. From block 202, the method advances to block 204 and starts a timer. The timer may be set for a predetermined time period such as ten seconds to carry out the commands as illustrated in FIG. 4. The user 12 receives verification from the system 10 when the commands are completely performed. The user 12 determines whether the commands are completed within the predetermined time period. After block 204, the method advances to bubble 206 and ends. It should be appreciated that branching opportunities, reflecting testing multiple possibilities based on various status conditions, is also implemented. It should also be appreciated that controllers test conditions from resources or locations. It should be appreciated that the user can force conditions in the operator logic that allows diagnostic conditions to be verified.] (FIGS. 3 and 4; Specification, page 10, line 8 through page 11, line 9).

The method still further includes the steps of loading the PLC logic in the PLC controlling the workcell if the PLC logic for the workcell is correct and using the PLC logic by

the PLC to operate the workcell. [Once the PLC code is verified, it is exported by the computer 14 via an electronic link to at least one PLC 20. The PLC 20 is then used at physical tool build to produce or build a workcell (not shown), which is used in a tooling line (not shown) for the manufacture of parts (not shown) for a motor vehicle (not shown).] (FIG. 1; Specification, page 7, lines 3 through 8).

Independent claim 9

The claimed subject matter of independent claim 9 is directed to a method of logical modeling operator interaction with a programmable logic controller logic verification system. [Referring to FIGS. 2 through 5, a method, according to the present invention, for logical modeling of operator interaction with the PLC logical verification system 16 of the system 10 is shown. In general, the user 12 models a human operator (not shown) in context of the part or product being developed, as a controller with a unique set of resources. In the present invention, a controller may be an operator, robot, PLC, or any programmable device. The resource assigned to the operator controller is a model of a physical manifestation of the operator in the workcell. For example, an operator resource for moving a part in a repetitive cycle might be an overhead crane that they directly control. Once the resource is bound to the operator controller, the PLC code or controller program can be written using conventional logic.] (FIGS. 2 through 5; Specification, page 7, line 15 through page 8, line 6).

The method includes the steps of constructing a flowchart that describes a series of commands for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent. [Referring to FIG. 2, the method begins in bubble 100 and advances to

block 102. In block 102, the method includes selecting "commands" from a resource's capability to cause an action. The commands available to the operator flowchart (selected while the user 12 is in the action block) are determined by the collection of resources that have been bound to that controller. For example, a resource may be "Carry" for an operator to carry a workpiece from a first location to a second location. The resource has at least one, preferably a plurality of capabilities. For example, a capability for the resource "carry" may be "lift" such that the operator lifts the workpiece before carrying the workpiece from the first location to the second location. It should also be appreciated that the method uses flowcharts to represent the cyclic logical behavior of the operator. It should further be appreciated that the purpose of the model is to verify the PLC logic by providing input signals to the PLC at desired times or based on conditional events.] (FIG. 2; Specification, page 9, lines 10 through 22 and page 9, lines 2 through 6).

The method also includes the steps of modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart. [The method begins by writing a control model for an operator by the operator interaction design system 18. For example, the operator interaction design system 18 will create a control model definition that describes how an operator picks up a part, carries, and loads it into a fixture. As illustrated in FIG. 5, a flowchart of a method for a part flow in two stations or workcells is shown. The method begins in bubble 50 where an operator or pusher moves the part to Station 0 and on a transfer bar in block 52. The part moves forward in block 54, moves up in block 56, until the part is at a top in block 58. The part drops to the transfer bar in block 60. The part is at workspace 1 in block 62. The part moves forward to Station 1 in block 64. The part moves up in block 66, until the part is at the top in block 68. The part drops to a second transfer bar in block 70. The part is at workspace 2

in block 72. It should be appreciated that the control model is information that describes events, dependencies, and logical conditions. It should also be appreciated that the method uses flowcharts to represent the cyclic logical behavior of the operator. It should further be appreciated that the purpose of the model is to verify the PLC logic by providing input signals to the PLC at desired times or based on conditional events. It should also be appreciated that the focus is on the logical representation of the operator and not the visual or spatial representations.] (FIGS. 2 and 5; Specification, page 8, lines 7 through page 9, line 9).

The method includes the steps of testing the control model by starting a timer and executing the commands by a PLC logical verification system on the computer to test whether PLC logic for the workcell is correct. [Referring to FIG. 3, the method allows either the operator logic or PLC to test for status being returned from a resource through its input registers. The execution of the flowchart during simulation, which might be more than one based on different starting conditions, then proceeds by successive "command, test status" loops. The method begins with initializing the test in bubble 200. From bubble 200, the method advances to block 202 and idles the operator. For example, in block 202, the operator is set to idle where no work or motion is being performed. From block 202, the method advances to block 204 and starts a timer. The timer may be set for a predetermined time period such as ten seconds to carry out the commands as illustrated in FIG. 4. The user 12 receives verification from the system 10 when the commands are completely performed. The user 12 determines whether the commands are completed within the predetermined time period. After block 204, the method advances to bubble 206 and ends. It should be appreciated that branching opportunities, reflecting testing multiple possibilities based on various status conditions, is also implemented. It should also be appreciated that controllers test conditions from resources or locations. It should be appreciated that the user can force conditions in the operator logic that allows diagnostic conditions to be

verified.] (FIGS. 3 and 4; Specification, page 10, line 8 through page 11, line 9).

The method further includes the steps of loading the PLC logic in the PLC controlling the workcell if the PLC logic is correct and using the PLC logic by the PLC to operate the workcell. [Once the PLC code is verified, it is exported by the computer 14 via an electronic link to at least one PLC 20. The PLC 20 is then used at physical tool build to produce or build a workcell (not shown), which is used in a tooling line (not shown) for the manufacture of parts (not shown) for a motor vehicle (not shown).] (FIG. 1; Specification, page 7, lines 3 through 8).

Independent claim 15

The claimed subject matter of independent claim 15 is directed to a method of logical modeling operator interaction with a programmable logic controller logic verification system. [Referring to FIGS. 2 through 5, a method, according to the present invention, for logical modeling of operator interaction with the PLC logical verification system 16 of the system 10 is shown. In general, the user 12 models a human operator (not shown) in context of the part or product being developed, as a controller with a unique set of resources. In the present invention, a controller may be an operator, robot, PLC, or any programmable device. The resource assigned to the operator controller is a model of a physical manifestation of the operator in the workcell. For example, an operator resource for moving a part in a repetitive cycle might be an overhead crane that they directly control. Once the resource is bound to the operator controller, the PLC code or controller program can be written using conventional logic.] (FIGS. 2 through 5; Specification, page 7, line 15 through page 8, line 6).

The method includes the steps of constructing a flowchart of a series of commands having at least one resource with at least one capability for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent. [Referring to FIG. 2, the method begins in bubble 100 and advances to block 102. In block 102, the method includes selecting "commands" from a resource's capability to cause an action. The commands available to the operator flowchart (selected while the user 12 is in the action block) are determined by the collection of resources that have been bound to that controller. For example, a resource may be "Carry" for an operator to carry a workpiece from a first location to a second location. The resource has at least one, preferably a plurality of capabilities. For example, a capability for the resource "carry" may be "lift" such that the operator lifts the workpiece before carrying the workpiece from the first location to the second location. The flowchart evokes resources and the user 12 can test the flowchart as to what is being tested is a PLC program loaded to the controller.] (FIG. 2; Specification, page 9, lines 10 through 22 and page 11, lines 7 through 9).

The method also includes the steps of modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart. [The method begins by writing a control model for an operator by the operator interaction design system 18. For example, the operator interaction design system 18 will create a control model definition that describes how an operator picks up a part, carries, and loads it into a fixture. As illustrated in FIG. 5, a flowchart of a method for a part flow in two stations or workcells is shown. The method begins in bubble 50 where an operator or pusher moves the part to Station 0 and on a transfer bar in block 52. The part moves forward in block 54, moves up in block 56, until the part is at a top in block 58. The

part drops to the transfer bar in block 60. The part is at workspace 1 in block 62. The part moves forward to Station 1 in block 64. The part moves up in block 66, until the part is at the top in block 68. The part drops to a second transfer bar in block 70. The part is at workspace 2 in block 72. It should be appreciated that the control model is information that describes events, dependencies, and logical conditions. It should also be appreciated that the method uses flowcharts to represent the cyclic logical behavior of the operator. It should further be appreciated that the purpose of the model is to verify the PLC logic by providing input signals to the PLC at desired times or based on conditional events.] (FIGS. 2 and 5; Specification, page 8, lines 7 through page 9, line 6).

The method includes the steps of initializing the operator interaction and idling the operator and testing the control model by starting a timer, executing the commands by a PLC logical verification system on the computer, and determining whether the commands are completed within a predetermined time to test whether PLC logic for the workcell is correct. [Referring to FIG. 3, the method allows either the operator logic or PLC to test for status being returned from a resource through its input registers. The execution of the flowchart during simulation, which might be more than one based on different starting conditions, then proceeds by successive "command, test status" loops. The method begins with initializing the test in bubble 200. From bubble 200, the method advances to block 202 and idles the operator. For example, in block 202, the operator is set to idle where no work or motion is being performed. From block 202, the method advances to block 204 and starts a timer. The timer may be set for a predetermined time period such as ten seconds to carry out the commands as illustrated in FIG. 4. The user 12 receives verification from the system 10 when the commands are completely performed. The user 12 determines whether the commands are completed within the predetermined time period. After block 204, the method advances to bubble 206 and ends. It

should be appreciated that branching opportunities, reflecting testing multiple possibilities based on various status conditions, is also implemented. It should also be appreciated that controllers test conditions from resources or locations. It should be appreciated that the user can force conditions in the operator logic that allows diagnostic conditions to be verified.] (FIG. 3; Specification, page 10, line 8 through page 11, line 6).

The method further includes the steps of loading the PLC logic in the PLC controlling the workcell if the PLC logic is correct and using the PLC logic by the PLC to operate the workcell. [Once the PLC code is verified, it is exported by the computer 14 via an electronic link to at least one PLC 20. The PLC 20 is then used at physical tool build to produce or build a workcell (not shown), which is used in a tooling line (not shown) for the manufacture of parts (not shown) for a motor vehicle (not shown).] (FIG. 1; Specification, page 7, lines 3 through 8).

GROUND OF REJECTION TO BE REVIEWED ON APPEAL

The ground of rejection to be reviewed on appeal is whether the claimed invention of claims 1 through 15 is obvious and unpatentable under 35 U.S.C. § 103 over "Handbook of Simulation" edited by Jerry Banks in view of "Simulation Modeling with Event Graphs" by Lee Schruben.

ARGUMENT

Claims Not Obvious or Unpatentable Under 35 U.S.C. § 103

As to patentability, 35 U.S.C. § 103 provides that a patent may not be obtained:

> If the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Id.

The United States Supreme Court interpreted the standard for 35 U.S.C. § 103 in Graham v. John Deere, 383 U.S. 1, 148 U.S.P.Q. 459 (1966). In Graham, the Court stated that under 35 U.S.C. § 103:

> The scope and content of the prior art are to be determined; differences between the prior art and the claims at issue are to be ascertained; and the level of ordinary skill in the pertinent art resolved. Against this background, the obviousness or non-obviousness of the subject matter is determined. 148 U.S.P.Q. at 467.

Using the standard set forth in Graham, the scope and content of the prior art relied upon by the Examiner will be determined.

As to the primary reference applied by the Examiner, the publication of "Handbook of Simulation", edited by Jerry Banks, discloses principles, methodology, advances, applications, and practice. An entity represents an object that requires explicit definition. An entity can be dynamic in that it "moves" through the system, or it can be static in that it serves other entities. An entity may have attributes that pertain to that entity alone. Thus attributes should be considered local values. A resource is an entity that provides service to dynamic entities. The resource can serve one or more than one dynamic entity at the same time (i.e., operate as a parallel server). A dynamic entity can request one or more units of a resource. Verification concerns the operational model. Is it performing properly? Validation is the determination that the conceptual model is an accurate representation of the real system. If the

client has been involved throughout the study period, and the simulation analyst has followed all the steps rigorously, the likelihood of a successful implementation is increased.

As to the secondary reference applied by the Examiner, the publication of "Simulation Modeling with Event Graphs" by Lee Schruben discloses that an event graph can be used to develop alternative event-oriented representations of a system. Several candidate model structures can be considered for possible implementation as discrete-event simulations using an event-scheduling approach. Applications of basic graph analysis techniques are illustrated in the context of two examples. Event graph analysis can aid in identifying state variables, in determining what events must be initially scheduled, in anticipating possible logic errors due to simultaneous events, and in eliminating unnecessary event routines prior to coding a simulation..

Claims 1 through 8

In contradistinction, claim 1 claims the present invention claimed as a method of logical modeling operator interaction with a programmable logic controller logical verification system. The method includes the steps of constructing a flowchart that describes interaction of an operator in a workcell using a computer wherein such interaction comprises sequential operations and asynchronous operations, the asynchronous operations being not time dependent. The method also includes the steps of modeling the operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on predefined conditions described in the flowchart. The method includes the steps of testing the control model by a PLC logical verification system on the computer as to whether PLC logic for the workcell is correct. The method further includes the steps of loading the PLC logic in the

PLC controlling the workcell if the PLC logic for the workcell is correct and using the PLC logic by the PLC to operate the workcell.

As to the differences between the prior art and the claims at issue, Banks merely discloses a handbook of simulation in which an entity can be dynamic in that it "moves" through the system, verification of an operational model, and validation of the conceptual model being an accurate representation of the real system. Banks lacks constructing a flowchart that describes interaction of an operator in a workcell using a computer wherein such interaction comprises sequential operations and asynchronous operations, the asynchronous operations being not time dependent, and modeling the operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on predefined conditions described in the flowchart. Banks also lacks testing the control model by a PLC logical verification system on the computer as to whether PLC logic for the workcell is correct and loading the PLC logic in the PLC controlling the workcell if the PLC logic for the workcell is correct. In Banks, there is no logical modeling of operator interaction with a programmable logic controller logical verification system and there are no asynchronous operations of the operator. Also in Banks, there is no modeling of an operator as an input to a programmable logic controller (PLC). Further, Banks is not used to debug PLC logic.

Schruben merely discloses that an event graph can be used to develop alternative event-oriented representations of a system in which several candidate model structures can be considered for possible implementation as discrete-event simulations using an event-scheduling approach. Schruben lacks constructing a flowchart that describes interaction of an operator in a workcell using a computer wherein such interaction comprises sequential operations and asynchronous operations, the asynchronous operations being not time dependent. In Schruben, there are discrete event simulations, which are time based, and cannot account for asynchronous

operations. For example, the (random) time required to repair a jammed machine is modeled as a discrete or time based event because it is denoted by "t" and, therefore, cannot be an asynchronous operation. The passages from Schruben cited by the Examiner do not constitute constructing a flowchart that describes interaction of an operator in a workcell using a computer wherein such interaction comprises sequential operations and asynchronous operations, the asynchronous operations being not time dependent. In Schruben, an operator may be responsible for loading and unloading parts that are processed by a machine as well as freeing a jammed machine, but this operator interaction is <u>not</u> modeled by a flowchart. Further, there is no modeling of an operator as an input to a programmable logic controller (PLC).

As to the level of ordinary skill in the pertinent art, in Banks, an entity can be dynamic in that it "moves" through the system. In Schruben, discrete event simulations are used for time dependent events and do not allow for asynchronous operations, which are <u>not</u> time dependent. Further, neither reference allows for modeling of an operator as an input to a programmable logic controller (PLC). As such, there is absolutely <u>no teaching</u> of a level of skill in the programmable logic controller art that a method of logical modeling operator interaction with a programmable logic controller logical verification system includes the steps of constructing a flowchart that describes interaction of an operator in a workcell using a computer wherein such interaction comprises sequential operations and asynchronous operations, the asynchronous operations being not time dependent, modeling the operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on predefined conditions described in the flowchart, and testing the control model by a PLC logical verification system on the computer as to whether PLC logic for the workcell is correct. The Examiner may not, because he doubts that the invention is patentable,

resort to speculation, unfounded assumptions or hindsight reconstruction to supply deficiencies in the factual basis. See In re Warner, 379 F. 2d 1011, 154 U.S.P.Q. 173 (C.C.P.A. 1967).

Even if the references could be combined, they do not teach a level of skill in the art of programmable logic controller of logical modeling operator interaction with a programmable logic controller logical verification system includes the steps of modeling the operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on predefined conditions described in the flowchart. Applicants are not attacking the references individually, but are clearly pointing out that each reference is deficient and, if combined (although Applicants maintain that they are not combinable), the combination is deficient. The present invention sets forth a unique and non-obvious combination of a method for logical modeling of operator interaction with a programmable logic controller logical verification system that allows a user to verify that the PLC code being planned will work as intended, prior to physically building the tools/manufacturing line and locating equipment. Unlike the prior art, the focus of the present invention is on the logical representation of the operator and not the visual or spatial representations of the operator. Contrary to the Examiner, this is reflected in the claim language because it recites the step of modeling the operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on predefined conditions described in the flowchart. Further, the additional claim language of "the asynchronous operations being not time dependent" was stated by the Examiner in the Advisory Action to overcome the Schruben's reference description of randomly occurring events by explicitly excluding any "time dependence" for "asynchronous operations".

In addition, the Examiner has adduced no factual basis to support his/her position that it would have been obvious to one of ordinary skill in the art to include the operator and

operator's interaction with the PLC-controlled machinery in the PLC logical verification system taught by Banks and that the modification could comprise a "simulated operator" pushing a START or RESTART button on a PLC-controlled machine after "freeing a jammed machine". Thus, the Examiner's stated conclusion of obviousness is based on speculation, unfounded assumptions or hindsight reconstruction to supply deficiencies in the factual basis.

The references, if combinable, fail to teach or suggest the combination of a method of logical modeling operator interaction with a programmable logic controller logical verification system including the steps of constructing a flowchart that describes interaction of an operator in a workcell using a computer wherein such interaction comprises sequential operations and asynchronous operations, the asynchronous operations being not time dependent, modeling the operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on predefined conditions described in the flowchart, testing the control model by a PLC logical verification system on the computer as to whether PLC logic for the workcell is correct, loading the PLC logic in the PLC controlling the workcell if the PLC logic for the workcell is correct, and using the PLC logic by the PLC to operate the workcell as claimed by Applicants. Thus, the Examiner has failed to establish a case of prima facie obviousness.

Against this background, it is submitted that the present invention of claims 1 through 8 is not obvious in view of a proposed combination of Banks and Schruben. The references cannot be combined to render obvious the combination of the method of logical modeling operator interaction with a programmable logic controller logical verification system of claims 1 through 8. Therefore, it is respectfully submitted that claims 1 through 8 are not obvious and are allowable over the rejection under 35 U.S.C. § 103.

Claims 9 through 14

As to claim 9, claim 9 claims the present invention as a method of logical modeling operator interaction with a programmable logic controller logic verification system. The method includes the steps of constructing a flowchart that describes a series of commands for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent. The method also includes the steps of modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart. The method includes the steps of testing the control model by starting a timer and executing the commands by a PLC logical verification system on the computer to test whether PLC logic for the workcell is correct. The method further includes the steps of loading the PLC logic in the PLC controlling the workcell if the PLC logic is correct and using the PLC logic by the PLC to operate the workcell.

As to the differences between the prior art and the claims at issue, Banks merely discloses a handbook of simulation in which an entity can be dynamic in that it "moves" through the system, verification of an operational model, and validation of the conceptual model being an accurate representation of the real system. Banks lacks constructing a flowchart that describes a series of commands for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent, and modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart. Banks also lacks testing the control model by a starting a timer and executing the commands by a PLC logical verification system on the

computer as to test whether PLC logic for the workcell is correct and loading the PLC logic in the PLC controlling the workcell if the PLC logic for the workcell is correct. In Banks, there is no logical modeling of operator interaction with a programmable logic controller logical verification system and there are no asynchronous operations of the operator. Also in Banks, there is no modeling of a human operator as an input to a programmable logic controller (PLC). Further, Banks is not used to debug PLC logic.

Schruben merely discloses that an event graph can be used to develop alternative event-oriented representations of a system in which several candidate model structures can be considered for possible implementation as discrete-event simulations using an event-scheduling approach. Schruben lacks constructing a flowchart that describes a series of commands for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent. In Schruben, there are discrete event simulations, which are time based, and cannot account for asynchronous operations. For example, the (random) time required to repair a jammed machine is modeled as a discrete or time based event because it is denoted by "t" and, therefore, cannot be an asynchronous operation. The passages from Schruben cited by the Examiner do not constitute constructing a flowchart that describes interaction of a human operator in a workcell using a computer wherein such interaction comprises sequential operations and asynchronous operations. In Schruben, an operator may be responsible for loading and unloading parts that are processed by a machine as well as freeing a jammed machine, but this operator interaction is not modeled by a flowchart. Further, there is no modeling of a human operator as an input to a programmable logic controller (PLC).

As to the level of ordinary skill in the pertinent art, there is absolutely no teaching of a level of skill in the programmable logic controller art that a method of logical modeling

operator interaction with a programmable logic controller logical verification system includes the steps of constructing a flowchart that describes a series of commands for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent, modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart, and testing the control model by starting a timer and executing the commands by a PLC logical verification system on the computer as to whether PLC logic for the workcell is correct. The Examiner may not, because he doubts that the invention is patentable, resort to speculation, unfounded assumptions or hindsight reconstruction to supply deficiencies in the factual basis. See In re Warner, 379 F. 2d 1011, 154 U.S.P.Q. 173 (CCPA 1967).

The present invention sets forth a unique and non-obvious combination of a method for logical modeling of operator interaction with a programmable logic controller logical verification system that allows a user to verify that the PLC code being planned will work as intended, prior to physically building the tools/manufacturing line and locating equipment. Unlike the prior art, the focus of the present invention is on the logical representation of the operator and not the visual or spatial representations of the operator. Contrary to the Examiner, this is reflected in the claim language because it recites the step of modeling the operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on predefined conditions described in the flowchart. Further, the additional claim language of "the asynchronous operations being not time dependent" was stated by the Examiner in the Advisory Action to overcome the Schruben's reference description of randomly occurring events by explicitly excluding any "time dependence" for "asynchronous operations".

In addition, the Examiner has adduced no factual basis to support his/her position that it would have been obvious to one of ordinary skill in the art to include the operator and operator's interaction with the PLC-controlled machinery in the PLC logical verification system taught by Banks and that the modification could comprise a "simulated operator" pushing a START or RESTART button on a PLC-controlled machine after "freeing a jammed machine". Thus, the Examiner's stated conclusion of obviousness is based on speculation, unfounded assumptions or hindsight reconstruction to supply deficiencies in the factual basis.

The references, if combinable, fail to teach or suggest the combination of a method of logical modeling operator interaction with a programmable logic controller logic verification system including the steps of constructing a flowchart that describes a series of commands for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent, modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart, testing the control model by starting a timer and executing the commands by a PLC logical verification system on the computer to test whether PLC logic for the workcell is correct, and loading the PLC logic in the PLC controlling the workcell if the PLC logic is correct and using the PLC logic by the PLC to operate the workcell as claimed by Applicants.

Obviousness under § 103 is a legal conclusion based on factual evidence (In re Fine, 837 F.2d 1071, 1073, 5 U.S.P.Q.2d 1596, 1598 (Fed. Cir. 1988), and the subjective opinion of the Examiner as to what is or is not obvious, without evidence in support thereof, does not suffice. Since the Examiner has not provided a sufficient factual basis which is supportive of his

position (see In re Warner, 379 F.2d 1011, 1017, 154 U.S.P.Q. 173, 178 (C.C.P.A. 1967), cert. Denied, 389 U.S. 1057 (1968)), the rejection of claims 9 through 14 is improper.

Against this background, it is submitted that the present invention of claims 9 through 14 is not obvious in view of a proposed combination of Banks and Schruben. The references cannot be combined to render obvious the combination of the method of logical modeling operator interaction with a programmable logic controller logic verification system of claims 9 through 14. Therefore, it is respectfully submitted that claims 9 through 14 are not obvious and are allowable over the rejection under 35 U.S.C. § 103.

Claim 15

As to claim 15, claim 15 claims a method of logical modeling operator interaction with a programmable logic controller logic verification system. The method includes the steps of constructing a flowchart of a series of commands having at least one resource with at least one capability for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent. The method also includes the steps of modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart. The method includes the steps of initializing the operator interaction, idling the operator, testing the control model by starting a timer, executing the commands by a PLC logical verification system on the computer, and determining whether the commands are completed within a predetermined time to test whether PLC logic for the workcell is correct. The method further includes the steps of loading

the PLC logic in the PLC controlling the workcell if the PLC logic is correct and using the PLC logic by the PLC to operate the workcell.

As to the differences between the prior art and the claims at issue, Banks merely discloses a handbook of simulation in which an entity can be dynamic in that it "moves" through the system, verification of an operational model, and validation of the conceptual model being an accurate representation of the real system. Banks lacks constructing a flowchart of a series of commands having at least one resource with at least one capability for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent, and modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart. Banks also lacks testing the control model by starting a timer, executing the commands by a PLC logical verification system on the computer, and determining whether the commands are completed within a predetermined time to test whether PLC logic for the workcell is correct. In Banks, there is no logical modeling of operator interaction with a programmable logic controller logical verification system and there are no asynchronous operations of the operator. Also in Banks, there is no modeling of a human operator as an input to a programmable logic controller (PLC). Further, Banks is not used to debug PLC logic.

Schruben merely discloses that an event graph can be used to develop alternative event-oriented representations of a system in which several candidate model structures can be considered for possible implementation as discrete-event simulations using an event-scheduling approach. Schruben lacks constructing a flowchart of a series of commands having at least one resource with at least one capability for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the

asynchronous operations being not time dependent. In Schruben, there are discrete event simulations, which are time based, and cannot account for asynchronous operations. For example, the (random) time required to repair a jammed machine is modeled as a discrete or time based event because it is denoted by "t" and, therefore, cannot be an asynchronous operation. The passages from Schruben cited by the Examiner do not constitute constructing a flowchart that describes interaction of an operator in a workcell using a computer wherein such interaction comprises sequential operations and asynchronous operations. In Schruben, an operator may be responsible for loading and unloading parts that are processed by a machine as well as freeing a jammed machine, but this operator interaction is not modeled by a flowchart. Further, there is no modeling of a human operator as an input to a programmable logic controller (PLC).

As to the level of ordinary skill in the pertinent art, there is absolutely no teaching of a level of skill in the programmable logic controller art that a method of logical modeling operator interaction with a programmable logic controller logical verification system includes the steps of constructing a flowchart of a series of commands having at least one resource with at least one capability for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent, modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart, and testing the control model by starting a timer, executing the commands by a PLC logical verification system on the computer, and determining whether the commands are completed within a predetermined time to test whether PLC logic for the workcell is correct. The Examiner may not, because he doubts that the invention is patentable, resort to speculation, unfounded assumptions or hindsight reconstruction

to supply deficiencies in the factual basis. See In re Warner, 379 F. 2d 1011, 154 U.S.P.Q. 173 (CCPA 1967).

Even if the references could be combined, they do not teach a level of skill in the art of programmable logic controller of logical modeling operator interaction with a programmable logic controller logical verification system includes the steps of modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart. Applicants are not attacking the references individually, but are clearly pointing out that each reference is deficient and, if combined (although Applicants maintain that they are not combinable), the combination is deficient. The present invention sets forth a unique and non-obvious combination of a method for logical modeling of operator interaction with a programmable logic controller logical verification system that allows a user to verify that the PLC code being planned will work as intended, prior to physically building the tools/manufacturing line and locating equipment. Unlike the prior art, the focus of the present invention is on the logical representation of the operator and not the visual or spatial representations of the operator. Contrary to the Examiner, this is reflected in the claim language because it recites the step of modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on predefined conditions described in the flowchart. Further, the additional claim language of "the asynchronous operations being not time dependent" was stated by the Examiner in the Advisory Action to overcome the Schruben's reference description of randomly occurring events by explicitly excluding any "time dependence" for "asynchronous operations".

In addition, the Examiner has adduced no factual basis to support his/her position that it would have been obvious to one of ordinary skill in the art to include the operator and

operator's interaction with the PLC-controlled machinery in the PLC logical verification system taught by Banks and that the modification could comprise a "simulated operator" pushing a START or RESTART button on a PLC-controlled machine after "freeing a jammed machine". Thus, the Examiner's stated conclusion of obviousness is based on speculation, unfounded assumptions or hindsight reconstruction to supply deficiencies in the factual basis.

The references, if combinable, fail to teach or suggest the combination of a method of logical modeling operator interaction with a programmable logic controller logic verification system including the steps of constructing a flowchart that describes a series of commands for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent, modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart, testing the control model by starting a timer and executing the commands by a PLC logical verification system on the computer to test whether PLC logic for the workcell is correct, and loading the PLC logic in the PLC controlling the workcell if the PLC logic is correct and using the PLC logic by the PLC to operate the workcell as claimed by Applicants.

Obviousness under § 103 is a legal conclusion based on factual evidence (In re Fine, 837 F.2d 1071, 1073, 5 U.S.P.Q.2d 1596, 1598 (Fed. Cir. 1988), and the subjective opinion of the Examiner as to what is or is not obvious, without evidence in support thereof, does not suffice. Since the Examiner has not provided a sufficient factual basis which is supportive of his position (see In re Warner, 379 F.2d 1011, 1017, 154 U.S.P.Q. 173, 178 (C.C.P.A. 1967), cert. Denied, 389 U.S. 1057 (1968)), the rejection of claims 15 through 20 is improper.

Against this background, it is submitted that the present invention of claim 15 is not obvious in view of a proposed combination of Banks and Schruben. The references cannot be combined to render obvious the combination of the method of logical modeling operator interaction with a programmable logic controller logic verification system of claim 15. Therefore, it is respectfully submitted that claim 15 is not obvious and is allowable over the rejection under 35 U.S.C. § 103.

CONCLUSION

In conclusion, it is respectfully submitted that the rejection of claims 1 through 15 is improper and should be reversed.

Respectfully submitted,

By: ____________________
Daniel H. Bliss
Registration No. 32,398

BLISS McGLYNN, P.C.
2075 West Big Beaver Road
Suite 600
Troy, Michigan 48084-3443
(248) 649-6090

Dated: July 16, 2007

Attorney Docket No.: 0693.00257
Ford Disclosure No.: 200-0665

CLAIMS APPENDIX

The claims on appeal are as follows:

1. A method of logical modeling operator interaction with a programmable logic controller logical verification system, said method comprising the steps of:

constructing a flowchart that describes interaction of an operator in a workcell using a computer wherein such interaction comprises sequential operations and asynchronous operations, the asynchronous operations being not time dependent;

modeling the operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on predefined conditions described in the flowchart;

testing the control model by a PLC logical verification system on the computer as to whether PLC logic for the workcell is correct; and

loading the PLC logic in the PLC controlling the workcell if the PLC logic for the workcell is correct and using the PLC logic by the PLC to operate the workcell.

2. A method as set forth in claim 1 wherein the step of testing comprises starting a timer and determining whether the operator interaction of the flowchart is completed within a predetermined time.

3. A method as set forth in claim 2 wherein the step of testing includes initializing the operator interaction of the flowchart prior to starting the timer.

4. A method as set forth in claim 3 wherein said step of testing includes idling the operator prior to starting the timer.

5. A method as set forth in claim 1 wherein said step of constructing comprises constructing a series of commands for the operator using the computer.

6. A method as set forth in claim 5 wherein the commands have at least one resource.

7. A method as set forth in claim 6 wherein the at least one resource has at least one capability.

8. A method as set forth in claim 1 wherein the step of testing includes executing the commands when a timer is started.

9. A method of logical modeling operator interaction with a programmable logic controller logic verification system, said method comprising the steps of:

constructing a flowchart that describes a series of commands for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent;

modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart;

testing the control model by starting a timer and executing the commands by a PLC logical verification system on the computer to test whether PLC logic for the workcell is correct; and

loading the PLC logic in the PLC controlling the workcell if the PLC logic is correct and using the PLC logic by the PLC to operate the workcell.

10. A method as set forth in claim 9 wherein the step of testing includes determining whether the commands of the flowchart are completed within a predetermined time.

11. A method as set forth in claim 10 wherein the step of testing includes initializing the operator interaction of the flowchart prior to starting the timer.

12. A method as set forth in claim 11 wherein said step of testing includes idling the operator prior to starting the timer.

13. A method as set forth in claim 9 wherein said step of constructing comprises constructing commands having at least one resource.

14. A method as set forth in claim 13 wherein the at least one resource has at least one capability.

15. A method of logical modeling operator interaction with a programmable logic controller logic verification system, said method comprising the steps of:

constructing a flowchart of a series of commands having at least one resource with at least one capability for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent;

modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart;

initializing the operator interaction and idling the operator;

testing the control model by starting a timer, executing the commands by a PLC logical verification system on the computer, and determining whether the commands are completed within a predetermined time to test whether PLC logic for the workcell is correct; and

loading the PLC logic in the PLC controlling the workcell if the PLC logic is correct and using the PLC logic by the PLC to operate the workcell.

EVIDENCE APPENDIX

None

RELATED PROCEEDINGS APPENDIX

None

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Art Unit:	2123	
Examiner:	J. Proctor	
Applicant(s):	J. G. Walacavage et al.	
Serial No.:	09/965,904	**APPEAL BRIEF**
Filing Date:	September 28, 2001	
For:	METHOD OF LOGICAL MODELING OF OPERATOR INTERACTION WITH PROGRAMMABLE LOGIC CONTROLLER LOGICAL VERIFICATION SYSTEM	

Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450

Sir:

By Notice of Appeal filed May 14, 2007, Applicants have appealed the Final Rejection dated February 14, 2007 and submit this brief in support of that appeal.

REAL PARTY IN INTEREST

The real party in interest is the Assignee, Ford Global Technologies, Inc.

RELATED APPEALS AND INTERFERENCES

There are no related appeals or interferences regarding the present application.

CERTIFICATE OF MAILING: (37 C.F.R. 1.8) I hereby certify that this paper (along with any paper referred to as being attached or enclosed) is being deposited with the U.S. Postal Service with sufficient postage as First Class mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, Virginia 22313-1450 on July 16, 2007, by ______.
Daniel H. Bliss

STATUS OF CLAIMS

Claims 1 through 15 have been rejected.

Claims 1 through 15 are being appealed.

STATUS OF AMENDMENTS

An Amendment Under 37 C.F.R. 1.116 was filed on April 16, 2007 in response to the Final Office Action dated February 14, 2007. An Advisory Action dated May 3, 2007 was issued and indicated that the Amendment under 37 C.F.R. 1.116 did not place the application in a condition for allowance. The Advisory Action stated that "Applicants may overcome the Schruben's reference description of randomly occurring events by amendments to the claim language that explicitly exclude any 'time dependence' for 'asynchronous operations'". Accordingly, a Second Amendment Under 37 C.F.R. 1.116 was filed on May 14, 2007 in response to the Final Office Action dated February 14, 2007 and Advisory Action dated May 3, 2007, that amended the independent claims to explicitly exclude any "time dependence" for "asynchronous operations". A second Advisory Action dated June 5, 2007 was issued and indicated that the Amendment under 37 C.F.R. 1.116 would be entered upon filing an appeal. A Notice of Appeal, along with the requisite fee, was filed on May 14, 2007. The Appeal Brief, along with the requisite fee, is submitted herewith.

SUMMARY OF THE CLAIMED SUBJECT MATTER

Independent Claim 1

The claimed subject matter of independent claim 1 is directed to a method of logical modeling operator interaction with a programmable logic controller logical verification system. [Referring to FIGS. 2 through 5, a method, according to the present invention, for logical modeling of operator interaction with the PLC logical verification system 16 of the system 10 is shown. In general, the user 12 models a human operator (not shown) in context of the part or product being developed, as a controller with a unique set of resources. In the present invention, a controller may be an operator, robot, PLC, or any programmable device. The resource assigned to the operator controller is a model of a physical manifestation of the operator in the workcell. For example, an operator resource for moving a part in a repetitive cycle might be an overhead crane that they directly control. Once the resource is bound to the operator controller, the PLC code or controller program can be written using conventional logic.] (FIGS. 2 through 5; Specification, page 7, line 15 through page 8, line 6).

The method includes the steps of constructing a flowchart that describes interaction of an operator in a workcell using a computer wherein such interaction comprises sequential operations and asynchronous operations, the asynchronous operations being not time dependent. [Referring to FIG. 2, the method begins in bubble 100 and advances to block 102. In block 102, the method includes selecting "commands" from a resource's capability to cause an action. The commands available to the operator flowchart (selected while the user 12 is in the action block) are determined by the collection of resources that have been bound to that controller. For example, a resource may be "Carry" for an operator to carry a workpiece from a

first location to a second location. The resource has at least one, preferably a plurality of capabilities. For example, a capability for the resource "carry" may be "lift" such that the operator lifts the workpiece before carrying the workpiece from the first location to the second location. It should also be appreciated that the method uses flowcharts to represent the cyclic logical behavior of the operator. It should further be appreciated that the purpose of the model is to verify the PLC logic by providing input signals to the PLC at desired times or based on conditional events.] (FIG. 2; Specification, page 9, lines 10 through 22 and page 9, lines 2 through 6).

The method also includes the steps of modeling the operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on predefined conditions described in the flowchart. [The method begins by writing a control model for an operator by the operator interaction design system 18. For example, the operator interaction design system 18 will create a control model definition that describes how an operator picks up a part, carries, and loads it into a fixture. As illustrated in FIG. 5, a flowchart of a method for a part flow in two stations or workcells is shown. The method begins in bubble 50 where an operator or pusher moves the part to Station 0 and on a transfer bar in block 52. The part moves forward in block 54, moves up in block 56, until the part is at a top in block 58. The part drops to the transfer bar in block 60. The part is at workspace 1 in block 62. The part moves forward to Station 1 in block 64. The part moves up in block 66, until the part is at the top in block 68. The part drops to a second transfer bar in block 70. The part is at workspace 2 in block 72. It should be appreciated that the control model is information that describes events, dependencies, and logical conditions. It should also be appreciated that the method uses flowcharts to represent the cyclic logical behavior of the operator. It should further be appreciated that the purpose of the model is to verify the PLC logic

by providing input signals to the PLC at desired times or based on conditional events. It should also be appreciated that the focus is on the logical representation of the operator and not the visual or spatial representations.] (FIGS. 2 and 5; Specification, page 8, lines 7 through page 9, line 9).

The method further includes the steps of testing the control model by a PLC logical verification system on the computer as to whether PLC logic for the workcell is correct. [Referring to FIG. 3, the method allows either the operator logic or PLC to test for status being returned from a resource through its input registers. The execution of the flowchart during simulation, which might be more than one based on different starting conditions, then proceeds by successive "command, test status" loops. The method begins with initializing the test in bubble 200. From bubble 200, the method advances to block 202 and idles the operator. For example, in block 202, the operator is set to idle where no work or motion is being performed. From block 202, the method advances to block 204 and starts a timer. The timer may be set for a predetermined time period such as ten seconds to carry out the commands as illustrated in FIG. 4. The user 12 receives verification from the system 10 when the commands are completely performed. The user 12 determines whether the commands are completed within the predetermined time period. After block 204, the method advances to bubble 206 and ends. It should be appreciated that branching opportunities, reflecting testing multiple possibilities based on various status conditions, is also implemented. It should also be appreciated that controllers test conditions from resources or locations. It should be appreciated that the user can force conditions in the operator logic that allows diagnostic conditions to be verified.] (FIGS. 3 and 4; Specification, page 10, line 8 through page 11, line 9).

The method still further includes the steps of loading the PLC logic in the PLC controlling the workcell if the PLC logic for the workcell is correct and using the PLC logic by

the PLC to operate the workcell. [Once the PLC code is verified, it is exported by the computer 14 via an electronic link to at least one PLC 20. The PLC 20 is then used at physical tool build to produce or build a workcell (not shown), which is used in a tooling line (not shown) for the manufacture of parts (not shown) for a motor vehicle (not shown).] (FIG. 1; Specification, page 7, lines 3 through 8).

Independent claim 9

The claimed subject matter of independent claim 9 is directed to a method of logical modeling operator interaction with a programmable logic controller logic verification system. [Referring to FIGS. 2 through 5, a method, according to the present invention, for logical modeling of operator interaction with the PLC logical verification system 16 of the system 10 is shown. In general, the user 12 models a human operator (not shown) in context of the part or product being developed, as a controller with a unique set of resources. In the present invention, a controller may be an operator, robot, PLC, or any programmable device. The resource assigned to the operator controller is a model of a physical manifestation of the operator in the workcell. For example, an operator resource for moving a part in a repetitive cycle might be an overhead crane that they directly control. Once the resource is bound to the operator controller, the PLC code or controller program can be written using conventional logic.] (FIGS. 2 through 5; Specification, page 7, line 15 through page 8, line 6).

The method includes the steps of constructing a flowchart that describes a series of commands for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent. [Referring to FIG. 2, the method begins in bubble 100 and advances to

block 102. In block 102, the method includes selecting "commands" from a resource's capability to cause an action. The commands available to the operator flowchart (selected while the user 12 is in the action block) are determined by the collection of resources that have been bound to that controller. For example, a resource may be "Carry" for an operator to carry a workpiece from a first location to a second location. The resource has at least one, preferably a plurality of capabilities. For example, a capability for the resource "carry" may be "lift" such that the operator lifts the workpiece before carrying the workpiece from the first location to the second location. It should also be appreciated that the method uses flowcharts to represent the cyclic logical behavior of the operator. It should further be appreciated that the purpose of the model is to verify the PLC logic by providing input signals to the PLC at desired times or based on conditional events.] (FIG. 2; Specification, page 9, lines 10 through 22 and page 9, lines 2 through 6).

The method also includes the steps of modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart. [The method begins by writing a control model for an operator by the operator interaction design system 18. For example, the operator interaction design system 18 will create a control model definition that describes how an operator picks up a part, carries, and loads it into a fixture. As illustrated in FIG. 5, a flowchart of a method for a part flow in two stations or workcells is shown. The method begins in bubble 50 where an operator or pusher moves the part to Station 0 and on a transfer bar in block 52. The part moves forward in block 54, moves up in block 56, until the part is at a top in block 58. The part drops to the transfer bar in block 60. The part is at workspace 1 in block 62. The part moves forward to Station 1 in block 64. The part moves up in block 66, until the part is at the top in block 68. The part drops to a second transfer bar in block 70. The part is at workspace 2

in block 72. It should be appreciated that the control model is information that describes events, dependencies, and logical conditions. It should also be appreciated that the method uses flowcharts to represent the cyclic logical behavior of the operator. It should further be appreciated that the purpose of the model is to verify the PLC logic by providing input signals to the PLC at desired times or based on conditional events. It should also be appreciated that the focus is on the logical representation of the operator and not the visual or spatial representations.] (FIGS. 2 and 5; Specification, page 8, lines 7 through page 9, line 9).

The method includes the steps of testing the control model by starting a timer and executing the commands by a PLC logical verification system on the computer to test whether PLC logic for the workcell is correct. [Referring to FIG. 3, the method allows either the operator logic or PLC to test for status being returned from a resource through its input registers. The execution of the flowchart during simulation, which might be more than one based on different starting conditions, then proceeds by successive "command, test status" loops. The method begins with initializing the test in bubble 200. From bubble 200, the method advances to block 202 and idles the operator. For example, in block 202, the operator is set to idle where no work or motion is being performed. From block 202, the method advances to block 204 and starts a timer. The timer may be set for a predetermined time period such as ten seconds to carry out the commands as illustrated in FIG. 4. The user 12 receives verification from the system 10 when the commands are completely performed. The user 12 determines whether the commands are completed within the predetermined time period. After block 204, the method advances to bubble 206 and ends. It should be appreciated that branching opportunities, reflecting testing multiple possibilities based on various status conditions, is also implemented. It should also be appreciated that controllers test conditions from resources or locations. It should be appreciated that the user can force conditions in the operator logic that allows diagnostic conditions to be

verified.] (FIGS. 3 and 4; Specification, page 10, line 8 through page 11, line 9).

The method further includes the steps of loading the PLC logic in the PLC controlling the workcell if the PLC logic is correct and using the PLC logic by the PLC to operate the workcell. [Once the PLC code is verified, it is exported by the computer 14 via an electronic link to at least one PLC 20. The PLC 20 is then used at physical tool build to produce or build a workcell (not shown), which is used in a tooling line (not shown) for the manufacture of parts (not shown) for a motor vehicle (not shown).] (FIG. 1; Specification, page 7, lines 3 through 8).

Independent claim 15

The claimed subject matter of independent claim 15 is directed to a method of logical modeling operator interaction with a programmable logic controller logic verification system. [Referring to FIGS. 2 through 5, a method, according to the present invention, for logical modeling of operator interaction with the PLC logical verification system 16 of the system 10 is shown. In general, the user 12 models a human operator (not shown) in context of the part or product being developed, as a controller with a unique set of resources. In the present invention, a controller may be an operator, robot, PLC, or any programmable device. The resource assigned to the operator controller is a model of a physical manifestation of the operator in the workcell. For example, an operator resource for moving a part in a repetitive cycle might be an overhead crane that they directly control. Once the resource is bound to the operator controller, the PLC code or controller program can be written using conventional logic.] (FIGS. 2 through 5; Specification, page 7, line 15 through page 8, line 6).

The method includes the steps of constructing a flowchart of a series of commands having at least one resource with at least one capability for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent. [Referring to FIG. 2, the method begins in bubble 100 and advances to block 102. In block 102, the method includes selecting "commands" from a resource's capability to cause an action. The commands available to the operator flowchart (selected while the user 12 is in the action block) are determined by the collection of resources that have been bound to that controller. For example, a resource may be "Carry" for an operator to carry a workpiece from a first location to a second location. The resource has at least one, preferably a plurality of capabilities. For example, a capability for the resource "carry" may be "lift" such that the operator lifts the workpiece before carrying the workpiece from the first location to the second location. The flowchart evokes resources and the user 12 can test the flowchart as to what is being tested is a PLC program loaded to the controller.] (FIG. 2; Specification, page 9, lines 10 through 22 and page 11, lines 7 through 9).

The method also includes the steps of modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart. [The method begins by writing a control model for an operator by the operator interaction design system 18. For example, the operator interaction design system 18 will create a control model definition that describes how an operator picks up a part, carries, and loads it into a fixture. As illustrated in FIG. 5, a flowchart of a method for a part flow in two stations or workcells is shown. The method begins in bubble 50 where an operator or pusher moves the part to Station 0 and on a transfer bar in block 52. The part moves forward in block 54, moves up in block 56, until the part is at a top in block 58. The

part drops to the transfer bar in block 60. The part is at workspace 1 in block 62. The part moves forward to Station 1 in block 64. The part moves up in block 66, until the part is at the top in block 68. The part drops to a second transfer bar in block 70. The part is at workspace 2 in block 72. It should be appreciated that the control model is information that describes events, dependencies, and logical conditions. It should also be appreciated that the method uses flowcharts to represent the cyclic logical behavior of the operator. It should further be appreciated that the purpose of the model is to verify the PLC logic by providing input signals to the PLC at desired times or based on conditional events.] (FIGS. 2 and 5; Specification, page 8, lines 7 through page 9, line 6).

The method includes the steps of initializing the operator interaction and idling the operator and testing the control model by starting a timer, executing the commands by a PLC logical verification system on the computer, and determining whether the commands are completed within a predetermined time to test whether PLC logic for the workcell is correct. [Referring to FIG. 3, the method allows either the operator logic or PLC to test for status being returned from a resource through its input registers. The execution of the flowchart during simulation, which might be more than one based on different starting conditions, then proceeds by successive "command, test status" loops. The method begins with initializing the test in bubble 200. From bubble 200, the method advances to block 202 and idles the operator. For example, in block 202, the operator is set to idle where no work or motion is being performed. From block 202, the method advances to block 204 and starts a timer. The timer may be set for a predetermined time period such as ten seconds to carry out the commands as illustrated in FIG. 4. The user 12 receives verification from the system 10 when the commands are completely performed. The user 12 determines whether the commands are completed within the predetermined time period. After block 204, the method advances to bubble 206 and ends. It

should be appreciated that branching opportunities, reflecting testing multiple possibilities based on various status conditions, is also implemented. It should also be appreciated that controllers test conditions from resources or locations. It should be appreciated that the user can force conditions in the operator logic that allows diagnostic conditions to be verified.] (FIG. 3; Specification, page 10, line 8 through page 11, line 6).

The method further includes the steps of loading the PLC logic in the PLC controlling the workcell if the PLC logic is correct and using the PLC logic by the PLC to operate the workcell. [Once the PLC code is verified, it is exported by the computer 14 via an electronic link to at least one PLC 20. The PLC 20 is then used at physical tool build to produce or build a workcell (not shown), which is used in a tooling line (not shown) for the manufacture of parts (not shown) for a motor vehicle (not shown).] (FIG. 1; Specification, page 7, lines 3 through 8).

GROUND OF REJECTION TO BE REVIEWED ON APPEAL

The ground of rejection to be reviewed on appeal is whether the claimed invention of claims 1 through 15 is obvious and unpatentable under 35 U.S.C. § 103 over "Handbook of Simulation" edited by Jerry Banks in view of "Simulation Modeling with Event Graphs" by Lee Schruben.

ARGUMENT

Claims Not Obvious or Unpatentable Under 35 U.S.C. § 103

As to patentability, 35 U.S.C. § 103 provides that a patent may not be obtained:

> If the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Id.

The United States Supreme Court interpreted the standard for 35 U.S.C. § 103 in Graham v. John Deere, 383 U.S. 1, 148 U.S.P.Q. 459 (1966). In Graham, the Court stated that under 35 U.S.C. § 103:

> The scope and content of the prior art are to be determined; differences between the prior art and the claims at issue are to be ascertained; and the level of ordinary skill in the pertinent art resolved. Against this background, the obviousness or non-obviousness of the subject matter is determined. 148 U.S.P.Q. at 467.

Using the standard set forth in Graham, the scope and content of the prior art relied upon by the Examiner will be determined.

As to the primary reference applied by the Examiner, the publication of "Handbook of Simulation", edited by Jerry Banks, discloses principles, methodology, advances, applications, and practice. An entity represents an object that requires explicit definition. An entity can be dynamic in that it "moves" through the system, or it can be static in that it serves other entities. An entity may have attributes that pertain to that entity alone. Thus attributes should be considered local values. A resource is an entity that provides service to dynamic entities. The resource can serve one or more than one dynamic entity at the same time (i.e., operate as a parallel server). A dynamic entity can request one or more units of a resource. Verification concerns the operational model. Is it performing properly? Validation is the determination that the conceptual model is an accurate representation of the real system. If the

client has been involved throughout the study period, and the simulation analyst has followed all the steps rigorously, the likelihood of a successful implementation is increased.

As to the secondary reference applied by the Examiner, the publication of "Simulation Modeling with Event Graphs" by Lee Schruben discloses that an event graph can be used to develop alternative event-oriented representations of a system. Several candidate model structures can be considered for possible implementation as discrete-event simulations using an event-scheduling approach. Applications of basic graph analysis techniques are illustrated in the context of two examples. Event graph analysis can aid in identifying state variables, in determining what events must be initially scheduled, in anticipating possible logic errors due to simultaneous events, and in eliminating unnecessary event routines prior to coding a simulation..

Claims 1 through 8

In contradistinction, claim 1 claims the present invention claimed as a method of logical modeling operator interaction with a programmable logic controller logical verification system. The method includes the steps of constructing a flowchart that describes interaction of an operator in a workcell using a computer wherein such interaction comprises sequential operations and asynchronous operations, the asynchronous operations being not time dependent. The method also includes the steps of modeling the operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on predefined conditions described in the flowchart. The method includes the steps of testing the control model by a PLC logical verification system on the computer as to whether PLC logic for the workcell is correct. The method further includes the steps of loading the PLC logic in the

PLC controlling the workcell if the PLC logic for the workcell is correct and using the PLC logic by the PLC to operate the workcell.

As to the differences between the prior art and the claims at issue, Banks merely discloses a handbook of simulation in which an entity can be dynamic in that it "moves" through the system, verification of an operational model, and validation of the conceptual model being an accurate representation of the real system. Banks lacks constructing a flowchart that describes interaction of an operator in a workcell using a computer wherein such interaction comprises sequential operations and asynchronous operations, the asynchronous operations being not time dependent, and modeling the operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on predefined conditions described in the flowchart. Banks also lacks testing the control model by a PLC logical verification system on the computer as to whether PLC logic for the workcell is correct and loading the PLC logic in the PLC controlling the workcell if the PLC logic for the workcell is correct. In Banks, there is no logical modeling of operator interaction with a programmable logic controller logical verification system and there are no asynchronous operations of the operator. Also in Banks, there is no modeling of an operator as an input to a programmable logic controller (PLC). Further, Banks is not used to debug PLC logic.

Schruben merely discloses that an event graph can be used to develop alternative event-oriented representations of a system in which several candidate model structures can be considered for possible implementation as discrete-event simulations using an event-scheduling approach. Schruben lacks constructing a flowchart that describes interaction of an operator in a workcell using a computer wherein such interaction comprises sequential operations and asynchronous operations, the asynchronous operations being not time dependent. In Schruben, there are discrete event simulations, which are time based, and cannot account for asynchronous

operations. For example, the (random) time required to repair a jammed machine is modeled as a discrete or time based event because it is denoted by "t" and, therefore, cannot be an asynchronous operation. The passages from Schruben cited by the Examiner do not constitute constructing a flowchart that describes interaction of an operator in a workcell using a computer wherein such interaction comprises sequential operations and asynchronous operations, the asynchronous operations being not time dependent. In Schruben, an operator may be responsible for loading and unloading parts that are processed by a machine as well as freeing a jammed machine, but this operator interaction is not modeled by a flowchart. Further, there is no modeling of an operator as an input to a programmable logic controller (PLC).

As to the level of ordinary skill in the pertinent art, in Banks, an entity can be dynamic in that it "moves" through the system. In Schruben, discrete event simulations are used for time dependent events and do not allow for asynchronous operations, which are not time dependent. Further, neither reference allows for modeling of an operator as an input to a programmable logic controller (PLC). As such, there is absolutely no teaching of a level of skill in the programmable logic controller art that a method of logical modeling operator interaction with a programmable logic controller logical verification system includes the steps of constructing a flowchart that describes interaction of an operator in a workcell using a computer wherein such interaction comprises sequential operations and asynchronous operations, the asynchronous operations being not time dependent, modeling the operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on predefined conditions described in the flowchart, and testing the control model by a PLC logical verification system on the computer as to whether PLC logic for the workcell is correct. The Examiner may not, because he doubts that the invention is patentable,

resort to speculation, unfounded assumptions or hindsight reconstruction to supply deficiencies in the factual basis. See In re Warner, 379 F. 2d 1011, 154 U.S.P.Q. 173 (C.C.P.A. 1967).

Even if the references could be combined, they do not teach a level of skill in the art of programmable logic controller of logical modeling operator interaction with a programmable logic controller logical verification system includes the steps of modeling the operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on predefined conditions described in the flowchart. Applicants are not attacking the references individually, but are clearly pointing out that each reference is deficient and, if combined (although Applicants maintain that they are not combinable), the combination is deficient. The present invention sets forth a unique and non-obvious combination of a method for logical modeling of operator interaction with a programmable logic controller logical verification system that allows a user to verify that the PLC code being planned will work as intended, prior to physically building the tools/manufacturing line and locating equipment. Unlike the prior art, the focus of the present invention is on the logical representation of the operator and not the visual or spatial representations of the operator. Contrary to the Examiner, this is reflected in the claim language because it recites the step of modeling the operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on predefined conditions described in the flowchart. Further, the additional claim language of "the asynchronous operations being not time dependent" was stated by the Examiner in the Advisory Action to overcome the Schruben's reference description of randomly occurring events by explicitly excluding any "time dependence" for "asynchronous operations".

In addition, the Examiner has adduced no factual basis to support his/her position that it would have been obvious to one of ordinary skill in the art to include the operator and

operator's interaction with the PLC-controlled machinery in the PLC logical verification system taught by Banks and that the modification could comprise a "simulated operator" pushing a START or RESTART button on a PLC-controlled machine after "freeing a jammed machine". Thus, the Examiner's stated conclusion of obviousness is based on speculation, unfounded assumptions or hindsight reconstruction to supply deficiencies in the factual basis.

The references, if combinable, fail to teach or suggest the combination of a method of logical modeling operator interaction with a programmable logic controller logical verification system including the steps of constructing a flowchart that describes interaction of an operator in a workcell using a computer wherein such interaction comprises sequential operations and asynchronous operations, the asynchronous operations being not time dependent, modeling the operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on predefined conditions described in the flowchart, testing the control model by a PLC logical verification system on the computer as to whether PLC logic for the workcell is correct, loading the PLC logic in the PLC controlling the workcell if the PLC logic for the workcell is correct, and using the PLC logic by the PLC to operate the workcell as claimed by Applicants. Thus, the Examiner has failed to establish a case of <u>prima facie</u> obviousness.

Against this background, it is submitted that the present invention of <u>claims 1 through 8 is not obvious</u> in view of a proposed combination of Banks and Schruben. The references cannot be combined to render obvious the combination of the method of logical modeling operator interaction with a programmable logic controller logical verification system of claims 1 through 8. Therefore, it is respectfully submitted that <u>claims 1 through 8 are not obvious</u> and are allowable over the rejection under 35 U.S.C. § 103.

Claims 9 through 14

As to claim 9, claim 9 claims the present invention as a method of logical modeling operator interaction with a programmable logic controller logic verification system. The method includes the steps of constructing a flowchart that describes a series of commands for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent. The method also includes the steps of modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart. The method includes the steps of testing the control model by starting a timer and executing the commands by a PLC logical verification system on the computer to test whether PLC logic for the workcell is correct. The method further includes the steps of loading the PLC logic in the PLC controlling the workcell if the PLC logic is correct and using the PLC logic by the PLC to operate the workcell.

As to the differences between the prior art and the claims at issue, Banks merely discloses a handbook of simulation in which an entity can be dynamic in that it "moves" through the system, verification of an operational model, and validation of the conceptual model being an accurate representation of the real system. Banks lacks constructing a flowchart that describes a series of commands for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent, and modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart. Banks also lacks testing the control model by a starting a timer and executing the commands by a PLC logical verification system on the

computer as to test whether PLC logic for the workcell is correct and loading the PLC logic in the PLC controlling the workcell if the PLC logic for the workcell is correct. In Banks, there is no logical modeling of operator interaction with a programmable logic controller logical verification system and there are no asynchronous operations of the operator. Also in Banks, there is no modeling of a human operator as an input to a programmable logic controller (PLC). Further, Banks is <u>not</u> used to debug PLC logic.

Schruben <u>merely</u> discloses that an event graph can be used to develop alternative event-oriented representations of a system in which several candidate model structures can be considered for possible implementation as discrete-event simulations using an event-scheduling approach. Schruben lacks constructing a flowchart that describes a series of commands for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent. In Schruben, there are discrete event simulations, which are time based, and cannot account for asynchronous operations. For example, the (random) time required to repair a jammed machine is modeled as a discrete or time based event because it is denoted by "t" and, therefore, cannot be an asynchronous operation. The passages from Schruben cited by the Examiner do not constitute constructing a flowchart that describes interaction of a human operator in a workcell using a computer wherein such interaction comprises sequential operations and asynchronous operations. In Schruben, an operator may be responsible for loading and unloading parts that are processed by a machine as well as freeing a jammed machine, but this operator interaction is <u>not</u> modeled by a flowchart. Further, there is no modeling of a human operator as an input to a programmable logic controller (PLC).

As to the level of ordinary skill in the pertinent art, there is absolutely <u>no teaching</u> of a level of skill in the programmable logic controller art that a method of logical modeling

operator interaction with a programmable logic controller logical verification system includes the steps of constructing a flowchart that describes a series of commands for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent, modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart, and testing the control model by starting a timer and executing the commands by a PLC logical verification system on the computer as to whether PLC logic for the workcell is correct. The Examiner may not, because he doubts that the invention is patentable, resort to speculation, unfounded assumptions or hindsight reconstruction to supply deficiencies in the factual basis. See In re Warner, 379 F. 2d 1011, 154 U.S.P.Q. 173 (CCPA 1967).

The present invention sets forth a unique and non-obvious combination of a method for logical modeling of operator interaction with a programmable logic controller logical verification system that allows a user to verify that the PLC code being planned will work as intended, prior to physically building the tools/manufacturing line and locating equipment. Unlike the prior art, the focus of the present invention is on the logical representation of the operator and not the visual or spatial representations of the operator. Contrary to the Examiner, this is reflected in the claim language because it recites the step of modeling the operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on predefined conditions described in the flowchart. Further, the additional claim language of "the asynchronous operations being not time dependent" was stated by the Examiner in the Advisory Action to overcome the Schruben's reference description of randomly occurring events by explicitly excluding any "time dependence" for "asynchronous operations".

In addition, the Examiner has adduced no factual basis to support his/her position that it would have been obvious to one of ordinary skill in the art to include the operator and operator's interaction with the PLC-controlled machinery in the PLC logical verification system taught by Banks and that the modification could comprise a "simulated operator" pushing a START or RESTART button on a PLC-controlled machine after "freeing a jammed machine". Thus, the Examiner's stated conclusion of obviousness is based on speculation, unfounded assumptions or hindsight reconstruction to supply deficiencies in the factual basis.

The references, if combinable, fail to teach or suggest the combination of a method of logical modeling operator interaction with a programmable logic controller logic verification system including the steps of constructing a flowchart that describes a series of commands for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent, modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart, testing the control model by starting a timer and executing the commands by a PLC logical verification system on the computer to test whether PLC logic for the workcell is correct, and loading the PLC logic in the PLC controlling the workcell if the PLC logic is correct and using the PLC logic by the PLC to operate the workcell as claimed by Applicants.

Obviousness under § 103 is a legal conclusion based on factual evidence (In re Fine, 837 F.2d 1071, 1073, 5 U.S.P.Q.2d 1596, 1598 (Fed. Cir. 1988), and the subjective opinion of the Examiner as to what is or is not obvious, without evidence in support thereof, does not suffice. Since the Examiner has not provided a sufficient factual basis which is supportive of his

position (see In re Warner, 379 F.2d 1011, 1017, 154 U.S.P.Q. 173, 178 (C.C.P.A. 1967), cert. Denied, 389 U.S. 1057 (1968)), the rejection of claims 9 through 14 is improper.

Against this background, it is submitted that the present invention of claims 9 through 14 is not obvious in view of a proposed combination of Banks and Schruben. The references cannot be combined to render obvious the combination of the method of logical modeling operator interaction with a programmable logic controller logic verification system of claims 9 through 14. Therefore, it is respectfully submitted that claims 9 through 14 are not obvious and are allowable over the rejection under 35 U.S.C. § 103.

Claim 15

As to claim 15, claim 15 claims a method of logical modeling operator interaction with a programmable logic controller logic verification system. The method includes the steps of constructing a flowchart of a series of commands having at least one resource with at least one capability for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent. The method also includes the steps of modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart. The method includes the steps of initializing the operator interaction, idling the operator, testing the control model by starting a timer, executing the commands by a PLC logical verification system on the computer, and determining whether the commands are completed within a predetermined time to test whether PLC logic for the workcell is correct. The method further includes the steps of loading

the PLC logic in the PLC controlling the workcell if the PLC logic is correct and using the PLC logic by the PLC to operate the workcell.

As to the differences between the prior art and the claims at issue, Banks merely discloses a handbook of simulation in which an entity can be dynamic in that it "moves" through the system, verification of an operational model, and validation of the conceptual model being an accurate representation of the real system. Banks lacks constructing a flowchart of a series of commands having at least one resource with at least one capability for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent, and modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart. Banks also lacks testing the control model by starting a timer, executing the commands by a PLC logical verification system on the computer, and determining whether the commands are completed within a predetermined time to test whether PLC logic for the workcell is correct. In Banks, there is no logical modeling of operator interaction with a programmable logic controller logical verification system and there are no asynchronous operations of the operator. Also in Banks, there is no modeling of a human operator as an input to a programmable logic controller (PLC). Further, Banks is not used to debug PLC logic.

Schruben merely discloses that an event graph can be used to develop alternative event-oriented representations of a system in which several candidate model structures can be considered for possible implementation as discrete-event simulations using an event-scheduling approach. Schruben lacks constructing a flowchart of a series of commands having at least one resource with at least one capability for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the

asynchronous operations being not time dependent. In Schruben, there are discrete event simulations, which are time based, and cannot account for asynchronous operations. For example, the (random) time required to repair a jammed machine is modeled as a discrete or time based event because it is denoted by "t" and, therefore, cannot be an asynchronous operation. The passages from Schruben cited by the Examiner do not constitute constructing a flowchart that describes interaction of an operator in a workcell using a computer wherein such interaction comprises sequential operations and asynchronous operations. In Schruben, an operator may be responsible for loading and unloading parts that are processed by a machine as well as freeing a jammed machine, but this operator interaction is <u>not</u> modeled by a flowchart. Further, there is no modeling of a human operator as an input to a programmable logic controller (PLC).

As to the level of ordinary skill in the pertinent art, there is absolutely <u>no teaching</u> of a level of skill in the programmable logic controller art that a method of logical modeling operator interaction with a programmable logic controller logical verification system includes the steps of constructing a flowchart of a series of commands having at least one resource with at least one capability for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent, modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart, and testing the control model by starting a timer, executing the commands by a PLC logical verification system on the computer, and determining whether the commands are completed within a predetermined time to test whether PLC logic for the workcell is correct. The Examiner may not, because he doubts that the invention is patentable, resort to speculation, unfounded assumptions or hindsight reconstruction

to supply deficiencies in the factual basis. See In re Warner, 379 F. 2d 1011, 154 U.S.P.Q. 173 (CCPA 1967).

Even if the references could be combined, they do not teach a level of skill in the art of programmable logic controller of logical modeling operator interaction with a programmable logic controller logical verification system includes the steps of modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart. Applicants are not attacking the references individually, but are clearly pointing out that each reference is deficient and, if combined (although Applicants maintain that they are not combinable), the combination is deficient. The present invention sets forth a unique and non-obvious combination of a method for logical modeling of operator interaction with a programmable logic controller logical verification system that allows a user to verify that the PLC code being planned will work as intended, prior to physically building the tools/manufacturing line and locating equipment. Unlike the prior art, the focus of the present invention is on the logical representation of the operator and not the visual or spatial representations of the operator. Contrary to the Examiner, this is reflected in the claim language because it recites the step of modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on predefined conditions described in the flowchart. Further, the additional claim language of "the asynchronous operations being not time dependent" was stated by the Examiner in the Advisory Action to overcome the Schruben's reference description of randomly occurring events by explicitly excluding any "time dependence" for "asynchronous operations".

In addition, the Examiner has adduced no factual basis to support his/her position that it would have been obvious to one of ordinary skill in the art to include the operator and

operator's interaction with the PLC-controlled machinery in the PLC logical verification system taught by Banks and that the modification could comprise a "simulated operator" pushing a START or RESTART button on a PLC-controlled machine after "freeing a jammed machine". Thus, the Examiner's stated conclusion of obviousness is based on speculation, unfounded assumptions or hindsight reconstruction to supply deficiencies in the factual basis.

The references, if combinable, fail to teach or suggest the combination of a method of logical modeling operator interaction with a programmable logic controller logic verification system including the steps of constructing a flowchart that describes a series of commands for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent, modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart, testing the control model by starting a timer and executing the commands by a PLC logical verification system on the computer to test whether PLC logic for the workcell is correct, and loading the PLC logic in the PLC controlling the workcell if the PLC logic is correct and using the PLC logic by the PLC to operate the workcell as claimed by Applicants.

Obviousness under § 103 is a legal conclusion based on factual evidence (In re Fine, 837 F.2d 1071, 1073, 5 U.S.P.Q.2d 1596, 1598 (Fed. Cir. 1988), and the subjective opinion of the Examiner as to what is or is not obvious, without evidence in support thereof, does not suffice. Since the Examiner has not provided a sufficient factual basis which is supportive of his position (see In re Warner, 379 F.2d 1011, 1017, 154 U.S.P.Q. 173, 178 (C.C.P.A. 1967), cert. Denied, 389 U.S. 1057 (1968)), the rejection of claims 15 through 20 is improper.

Against this background, it is submitted that the present invention of claim 15 is not obvious in view of a proposed combination of Banks and Schruben. The references cannot be combined to render obvious the combination of the method of logical modeling operator interaction with a programmable logic controller logic verification system of claim 15. Therefore, it is respectfully submitted that claim 15 is not obvious and is allowable over the rejection under 35 U.S.C. § 103.

CONCLUSION

In conclusion, it is respectfully submitted that the rejection of claims 1 through 15 is improper and should be reversed.

Respectfully submitted,

By: Daniel H. Bliss

Daniel H. Bliss
Registration No. 32,398

BLISS McGLYNN, P.C.
2075 West Big Beaver Road
Suite 600
Troy, Michigan 48084-3443
(248) 649-6090

Dated: July 16, 2007

Attorney Docket No.: 0693.00257
Ford Disclosure No.: 200-0665

CLAIMS APPENDIX

The claims on appeal are as follows:

1. A method of logical modeling operator interaction with a programmable logic controller logical verification system, said method comprising the steps of:

constructing a flowchart that describes interaction of an operator in a workcell using a computer wherein such interaction comprises sequential operations and asynchronous operations, the asynchronous operations being not time dependent;

modeling the operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on predefined conditions described in the flowchart;

testing the control model by a PLC logical verification system on the computer as to whether PLC logic for the workcell is correct; and

loading the PLC logic in the PLC controlling the workcell if the PLC logic for the workcell is correct and using the PLC logic by the PLC to operate the workcell.

2. A method as set forth in claim 1 wherein the step of testing comprises starting a timer and determining whether the operator interaction of the flowchart is completed within a predetermined time.

3. A method as set forth in claim 2 wherein the step of testing includes initializing the operator interaction of the flowchart prior to starting the timer.

4. A method as set forth in claim 3 wherein said step of testing includes idling the operator prior to starting the timer.

5. A method as set forth in claim 1 wherein said step of constructing comprises constructing a series of commands for the operator using the computer.

6. A method as set forth in claim 5 wherein the commands have at least one resource.

7. A method as set forth in claim 6 wherein the at least one resource has at least one capability.

8. A method as set forth in claim 1 wherein the step of testing includes executing the commands when a timer is started.

9. A method of logical modeling operator interaction with a programmable logic controller logic verification system, said method comprising the steps of:

constructing a flowchart that describes a series of commands for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent;

modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart;

testing the control model by starting a timer and executing the commands by a PLC logical verification system on the computer to test whether PLC logic for the workcell is correct; and

loading the PLC logic in the PLC controlling the workcell if the PLC logic is correct and using the PLC logic by the PLC to operate the workcell.

10. A method as set forth in claim 9 wherein the step of testing includes determining whether the commands of the flowchart are completed within a predetermined time.

11. A method as set forth in claim 10 wherein the step of testing includes initializing the operator interaction of the flowchart prior to starting the timer.

12. A method as set forth in claim 11 wherein said step of testing includes idling the operator prior to starting the timer.

13. A method as set forth in claim 9 wherein said step of constructing comprises constructing commands having at least one resource.

14. A method as set forth in claim 13 wherein the at least one resource has at least one capability.

15. A method of logical modeling operator interaction with a programmable logic controller logic verification system, said method comprising the steps of:

constructing a flowchart of a series of commands having at least one resource with at least one capability for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent;

modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart;

initializing the operator interaction and idling the operator;

testing the control model by starting a timer, executing the commands by a PLC logical verification system on the computer, and determining whether the commands are completed within a predetermined time to test whether PLC logic for the workcell is correct; and

loading the PLC logic in the PLC controlling the workcell if the PLC logic is correct and using the PLC logic by the PLC to operate the workcell.

RELATED PROCEEDINGS APPENDIX

None

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Art Unit:	2123	
Examiner:	J. Proctor	
Applicant(s):	J. G. Walacavage et al.	
Serial No.:	09/965,904	**APPEAL BRIEF**
Filing Date:	September 28, 2001	
For:	METHOD OF LOGICAL MODELING OF OPERATOR INTERACTION WITH PROGRAMMABLE LOGIC CONTROLLER LOGICAL VERIFICATION SYSTEM	

Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450

Sir:

By Notice of Appeal filed May 14, 2007, Applicants have appealed the Final Rejection dated February 14, 2007 and submit this brief in support of that appeal.

REAL PARTY IN INTEREST

The real party in interest is the Assignee, Ford Global Technologies, Inc.

RELATED APPEALS AND INTERFERENCES

There are no related appeals or interferences regarding the present application.

CERTIFICATE OF MAILING: (37 C.F.R. 1.8) I hereby certify that this paper (along with any paper referred to as being attached or enclosed) is being deposited with the U.S. Postal Service with sufficient postage as First Class mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, Virginia 22313-1450 on July 16, 2007, by ________.
Daniel H. Bliss

STATUS OF CLAIMS

Claims 1 through 15 have been rejected.

Claims 1 through 15 are being appealed.

STATUS OF AMENDMENTS

An Amendment Under 37 C.F.R. 1.116 was filed on April 16, 2007 in response to the Final Office Action dated February 14, 2007. An Advisory Action dated May 3, 2007 was issued and indicated that the Amendment under 37 C.F.R. 1.116 did not place the application in a condition for allowance. The Advisory Action stated that "Applicants may overcome the Schruben's reference description of randomly occurring events by amendments to the claim language that explicitly exclude any 'time dependence' for 'asynchronous operations'". Accordingly, a Second Amendment Under 37 C.F.R. 1.116 was filed on May 14, 2007 in response to the Final Office Action dated February 14, 2007 and Advisory Action dated May 3, 2007, that amended the independent claims to explicitly exclude any "time dependence" for "asynchronous operations". A second Advisory Action dated June 5, 2007 was issued and indicated that the Amendment under 37 C.F.R. 1.116 would be entered upon filing an appeal. A Notice of Appeal, along with the requisite fee, was filed on May 14, 2007. The Appeal Brief, along with the requisite fee, is submitted herewith.

SUMMARY OF THE CLAIMED SUBJECT MATTER

Independent Claim 1

The claimed subject matter of independent claim 1 is directed to a method of logical modeling operator interaction with a programmable logic controller logical verification system. [Referring to FIGS. 2 through 5, a method, according to the present invention, for logical modeling of operator interaction with the PLC logical verification system 16 of the system 10 is shown. In general, the user 12 models a human operator (not shown) in context of the part or product being developed, as a controller with a unique set of resources. In the present invention, a controller may be an operator, robot, PLC, or any programmable device. The resource assigned to the operator controller is a model of a physical manifestation of the operator in the workcell. For example, an operator resource for moving a part in a repetitive cycle might be an overhead crane that they directly control. Once the resource is bound to the operator controller, the PLC code or controller program can be written using conventional logic.] (FIGS. 2 through 5; Specification, page 7, line 15 through page 8, line 6).

The method includes the steps of constructing a flowchart that describes interaction of an operator in a workcell using a computer wherein such interaction comprises sequential operations and asynchronous operations, the asynchronous operations being not time dependent. [Referring to FIG. 2, the method begins in bubble 100 and advances to block 102. In block 102, the method includes selecting "commands" from a resource's capability to cause an action. The commands available to the operator flowchart (selected while the user 12 is in the action block) are determined by the collection of resources that have been bound to that controller. For example, a resource may be "Carry" for an operator to carry a workpiece from a

first location to a second location. The resource has at least one, preferably a plurality of capabilities. For example, a capability for the resource "carry" may be "lift" such that the operator lifts the workpiece before carrying the workpiece from the first location to the second location. It should also be appreciated that the method uses flowcharts to represent the cyclic logical behavior of the operator. It should further be appreciated that the purpose of the model is to verify the PLC logic by providing input signals to the PLC at desired times or based on conditional events.] (FIG. 2; Specification, page 9, lines 10 through 22 and page 9, lines 2 through 6).

The method also includes the steps of modeling the operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on predefined conditions described in the flowchart. [The method begins by writing a control model for an operator by the operator interaction design system 18. For example, the operator interaction design system 18 will create a control model definition that describes how an operator picks up a part, carries, and loads it into a fixture. As illustrated in FIG. 5, a flowchart of a method for a part flow in two stations or workcells is shown. The method begins in bubble 50 where an operator or pusher moves the part to Station 0 and on a transfer bar in block 52. The part moves forward in block 54, moves up in block 56, until the part is at a top in block 58. The part drops to the transfer bar in block 60. The part is at workspace 1 in block 62. The part moves forward to Station 1 in block 64. The part moves up in block 66, until the part is at the top in block 68. The part drops to a second transfer bar in block 70. The part is at workspace 2 in block 72. It should be appreciated that the control model is information that describes events, dependencies, and logical conditions. It should also be appreciated that the method uses flowcharts to represent the cyclic logical behavior of the operator. It should further be appreciated that the purpose of the model is to verify the PLC logic

by providing input signals to the PLC at desired times or based on conditional events. It should also be appreciated that the focus is on the logical representation of the operator and not the visual or spatial representations.] (FIGS. 2 and 5; Specification, page 8, lines 7 through page 9, line 9).

The method further includes the steps of testing the control model by a PLC logical verification system on the computer as to whether PLC logic for the workcell is correct. [Referring to FIG. 3, the method allows either the operator logic or PLC to test for status being returned from a resource through its input registers. The execution of the flowchart during simulation, which might be more than one based on different starting conditions, then proceeds by successive "command, test status" loops. The method begins with initializing the test in bubble 200. From bubble 200, the method advances to block 202 and idles the operator. For example, in block 202, the operator is set to idle where no work or motion is being performed. From block 202, the method advances to block 204 and starts a timer. The timer may be set for a predetermined time period such as ten seconds to carry out the commands as illustrated in FIG. 4. The user 12 receives verification from the system 10 when the commands are completely performed. The user 12 determines whether the commands are completed within the predetermined time period. After block 204, the method advances to bubble 206 and ends. It should be appreciated that branching opportunities, reflecting testing multiple possibilities based on various status conditions, is also implemented. It should also be appreciated that controllers test conditions from resources or locations. It should be appreciated that the user can force conditions in the operator logic that allows diagnostic conditions to be verified.] (FIGS. 3 and 4; Specification, page 10, line 8 through page 11, line 9).

The method still further includes the steps of loading the PLC logic in the PLC controlling the workcell if the PLC logic for the workcell is correct and using the PLC logic by

the PLC to operate the workcell. [Once the PLC code is verified, it is exported by the computer 14 via an electronic link to at least one PLC 20. The PLC 20 is then used at physical tool build to produce or build a workcell (not shown), which is used in a tooling line (not shown) for the manufacture of parts (not shown) for a motor vehicle (not shown).] (FIG. 1; Specification, page 7, lines 3 through 8).

Independent claim 9

The claimed subject matter of independent claim 9 is directed to a method of logical modeling operator interaction with a programmable logic controller logic verification system. [Referring to FIGS. 2 through 5, a method, according to the present invention, for logical modeling of operator interaction with the PLC logical verification system 16 of the system 10 is shown. In general, the user 12 models a human operator (not shown) in context of the part or product being developed, as a controller with a unique set of resources. In the present invention, a controller may be an operator, robot, PLC, or any programmable device. The resource assigned to the operator controller is a model of a physical manifestation of the operator in the workcell. For example, an operator resource for moving a part in a repetitive cycle might be an overhead crane that they directly control. Once the resource is bound to the operator controller, the PLC code or controller program can be written using conventional logic.] (FIGS. 2 through 5; Specification, page 7, line 15 through page 8, line 6).

The method includes the steps of constructing a flowchart that describes a series of commands for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent. [Referring to FIG. 2, the method begins in bubble 100 and advances to

block 102. In block 102, the method includes selecting "commands" from a resource's capability to cause an action. The commands available to the operator flowchart (selected while the user 12 is in the action block) are determined by the collection of resources that have been bound to that controller. For example, a resource may be "Carry" for an operator to carry a workpiece from a first location to a second location. The resource has at least one, preferably a plurality of capabilities. For example, a capability for the resource "carry" may be "lift" such that the operator lifts the workpiece before carrying the workpiece from the first location to the second location. It should also be appreciated that the method uses flowcharts to represent the cyclic logical behavior of the operator. It should further be appreciated that the purpose of the model is to verify the PLC logic by providing input signals to the PLC at desired times or based on conditional events.] (FIG. 2; Specification, page 9, lines 10 through 22 and page 9, lines 2 through 6).

The method also includes the steps of modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart. [The method begins by writing a control model for an operator by the operator interaction design system 18. For example, the operator interaction design system 18 will create a control model definition that describes how an operator picks up a part, carries, and loads it into a fixture. As illustrated in FIG. 5, a flowchart of a method for a part flow in two stations or workcells is shown. The method begins in bubble 50 where an operator or pusher moves the part to Station 0 and on a transfer bar in block 52. The part moves forward in block 54, moves up in block 56, until the part is at a top in block 58. The part drops to the transfer bar in block 60. The part is at workspace 1 in block 62. The part moves forward to Station 1 in block 64. The part moves up in block 66, until the part is at the top in block 68. The part drops to a second transfer bar in block 70. The part is at workspace 2

in block 72. It should be appreciated that the control model is information that describes events, dependencies, and logical conditions. It should also be appreciated that the method uses flowcharts to represent the cyclic logical behavior of the operator. It should further be appreciated that the purpose of the model is to verify the PLC logic by providing input signals to the PLC at desired times or based on conditional events. It should also be appreciated that the focus is on the logical representation of the operator and not the visual or spatial representations.] (FIGS. 2 and 5; Specification, page 8, lines 7 through page 9, line 9).

The method includes the steps of testing the control model by starting a timer and executing the commands by a PLC logical verification system on the computer to test whether PLC logic for the workcell is correct. [Referring to FIG. 3, the method allows either the operator logic or PLC to test for status being returned from a resource through its input registers. The execution of the flowchart during simulation, which might be more than one based on different starting conditions, then proceeds by successive "command, test status" loops. The method begins with initializing the test in bubble 200. From bubble 200, the method advances to block 202 and idles the operator. For example, in block 202, the operator is set to idle where no work or motion is being performed. From block 202, the method advances to block 204 and starts a timer. The timer may be set for a predetermined time period such as ten seconds to carry out the commands as illustrated in FIG. 4. The user 12 receives verification from the system 10 when the commands are completely performed. The user 12 determines whether the commands are completed within the predetermined time period. After block 204, the method advances to bubble 206 and ends. It should be appreciated that branching opportunities, reflecting testing multiple possibilities based on various status conditions, is also implemented. It should also be appreciated that controllers test conditions from resources or locations. It should be appreciated that the user can force conditions in the operator logic that allows diagnostic conditions to be

verified.] (FIGS. 3 and 4; Specification, page 10, line 8 through page 11, line 9).

The method further includes the steps of loading the PLC logic in the PLC controlling the workcell if the PLC logic is correct and using the PLC logic by the PLC to operate the workcell. [Once the PLC code is verified, it is exported by the computer 14 via an electronic link to at least one PLC 20. The PLC 20 is then used at physical tool build to produce or build a workcell (not shown), which is used in a tooling line (not shown) for the manufacture of parts (not shown) for a motor vehicle (not shown).] (FIG. 1; Specification, page 7, lines 3 through 8).

Independent claim 15

The claimed subject matter of independent claim 15 is directed to a method of logical modeling operator interaction with a programmable logic controller logic verification system. [Referring to FIGS. 2 through 5, a method, according to the present invention, for logical modeling of operator interaction with the PLC logical verification system 16 of the system 10 is shown. In general, the user 12 models a human operator (not shown) in context of the part or product being developed, as a controller with a unique set of resources. In the present invention, a controller may be an operator, robot, PLC, or any programmable device. The resource assigned to the operator controller is a model of a physical manifestation of the operator in the workcell. For example, an operator resource for moving a part in a repetitive cycle might be an overhead crane that they directly control. Once the resource is bound to the operator controller, the PLC code or controller program can be written using conventional logic.] (FIGS. 2 through 5; Specification, page 7, line 15 through page 8, line 6).

The method includes the steps of constructing a flowchart of a series of commands having at least one resource with at least one capability for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent. [Referring to FIG. 2, the method begins in bubble 100 and advances to block 102. In block 102, the method includes selecting "commands" from a resource's capability to cause an action. The commands available to the operator flowchart (selected while the user 12 is in the action block) are determined by the collection of resources that have been bound to that controller. For example, a resource may be "Carry" for an operator to carry a workpiece from a first location to a second location. The resource has at least one, preferably a plurality of capabilities. For example, a capability for the resource "carry" may be "lift" such that the operator lifts the workpiece before carrying the workpiece from the first location to the second location. The flowchart evokes resources and the user 12 can test the flowchart as to what is being tested is a PLC program loaded to the controller.] (FIG. 2; Specification, page 9, lines 10 through 22 and page 11, lines 7 through 9).

The method also includes the steps of modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart. [The method begins by writing a control model for an operator by the operator interaction design system 18. For example, the operator interaction design system 18 will create a control model definition that describes how an operator picks up a part, carries, and loads it into a fixture. As illustrated in FIG. 5, a flowchart of a method for a part flow in two stations or workcells is shown. The method begins in bubble 50 where an operator or pusher moves the part to Station 0 and on a transfer bar in block 52. The part moves forward in block 54, moves up in block 56, until the part is at a top in block 58. The

part drops to the transfer bar in block 60. The part is at workspace 1 in block 62. The part moves forward to Station 1 in block 64. The part moves up in block 66, until the part is at the top in block 68. The part drops to a second transfer bar in block 70. The part is at workspace 2 in block 72. It should be appreciated that the control model is information that describes events, dependencies, and logical conditions. It should also be appreciated that the method uses flowcharts to represent the cyclic logical behavior of the operator. It should further be appreciated that the purpose of the model is to verify the PLC logic by providing input signals to the PLC at desired times or based on conditional events.] (FIGS. 2 and 5; Specification, page 8, lines 7 through page 9, line 6).

The method includes the steps of initializing the operator interaction and idling the operator and testing the control model by starting a timer, executing the commands by a PLC logical verification system on the computer, and determining whether the commands are completed within a predetermined time to test whether PLC logic for the workcell is correct. [Referring to FIG. 3, the method allows either the operator logic or PLC to test for status being returned from a resource through its input registers. The execution of the flowchart during simulation, which might be more than one based on different starting conditions, then proceeds by successive "command, test status" loops. The method begins with initializing the test in bubble 200. From bubble 200, the method advances to block 202 and idles the operator. For example, in block 202, the operator is set to idle where no work or motion is being performed. From block 202, the method advances to block 204 and starts a timer. The timer may be set for a predetermined time period such as ten seconds to carry out the commands as illustrated in FIG. 4. The user 12 receives verification from the system 10 when the commands are completely performed. The user 12 determines whether the commands are completed within the predetermined time period. After block 204, the method advances to bubble 206 and ends. It

should be appreciated that branching opportunities, reflecting testing multiple possibilities based on various status conditions, is also implemented. It should also be appreciated that controllers test conditions from resources or locations. It should be appreciated that the user can force conditions in the operator logic that allows diagnostic conditions to be verified.] (FIG. 3; Specification, page 10, line 8 through page 11, line 6).

The method further includes the steps of loading the PLC logic in the PLC controlling the workcell if the PLC logic is correct and using the PLC logic by the PLC to operate the workcell. [Once the PLC code is verified, it is exported by the computer 14 via an electronic link to at least one PLC 20. The PLC 20 is then used at physical tool build to produce or build a workcell (not shown), which is used in a tooling line (not shown) for the manufacture of parts (not shown) for a motor vehicle (not shown).] (FIG. 1; Specification, page 7, lines 3 through 8).

GROUND OF REJECTION TO BE REVIEWED ON APPEAL

The ground of rejection to be reviewed on appeal is whether the claimed invention of claims 1 through 15 is obvious and unpatentable under 35 U.S.C. § 103 over "Handbook of Simulation" edited by Jerry Banks in view of "Simulation Modeling with Event Graphs" by Lee Schruben.

ARGUMENT

Claims Not Obvious or Unpatentable Under 35 U.S.C. § 103

As to patentability, 35 U.S.C. § 103 provides that a patent may not be obtained:

> If the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Id.

The United States Supreme Court interpreted the standard for 35 U.S.C. § 103 in Graham v. John Deere, 383 U.S. 1, 148 U.S.P.Q. 459 (1966). In Graham, the Court stated that under 35 U.S.C. § 103:

> The scope and content of the prior art are to be determined; differences between the prior art and the claims at issue are to be ascertained; and the level of ordinary skill in the pertinent art resolved. Against this background, the obviousness or non-obviousness of the subject matter is determined. 148 U.S.P.Q. at 467.

Using the standard set forth in Graham, the scope and content of the prior art relied upon by the Examiner will be determined.

As to the primary reference applied by the Examiner, the publication of "Handbook of Simulation", edited by Jerry Banks, discloses principles, methodology, advances, applications, and practice. An entity represents an object that requires explicit definition. An entity can be dynamic in that it "moves" through the system, or it can be static in that it serves other entities. An entity may have attributes that pertain to that entity alone. Thus attributes should be considered local values. A resource is an entity that provides service to dynamic entities. The resource can serve one or more than one dynamic entity at the same time (i.e., operate as a parallel server). A dynamic entity can request one or more units of a resource. Verification concerns the operational model. Is it performing properly? Validation is the determination that the conceptual model is an accurate representation of the real system. If the

client has been involved throughout the study period, and the simulation analyst has followed all the steps rigorously, the likelihood of a successful implementation is increased.

As to the secondary reference applied by the Examiner, the publication of "Simulation Modeling with Event Graphs" by Lee Schruben discloses that an event graph can be used to develop alternative event-oriented representations of a system. Several candidate model structures can be considered for possible implementation as discrete-event simulations using an event-scheduling approach. Applications of basic graph analysis techniques are illustrated in the context of two examples. Event graph analysis can aid in identifying state variables, in determining what events must be initially scheduled, in anticipating possible logic errors due to simultaneous events, and in eliminating unnecessary event routines prior to coding a simulation..

Claims 1 through 8

In contradistinction, claim 1 claims the present invention claimed as a method of logical modeling operator interaction with a programmable logic controller logical verification system. The method includes the steps of constructing a flowchart that describes interaction of an operator in a workcell using a computer wherein such interaction comprises sequential operations and asynchronous operations, the asynchronous operations being not time dependent. The method also includes the steps of modeling the operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on predefined conditions described in the flowchart. The method includes the steps of testing the control model by a PLC logical verification system on the computer as to whether PLC logic for the workcell is correct. The method further includes the steps of loading the PLC logic in the

PLC controlling the workcell if the PLC logic for the workcell is correct and using the PLC logic by the PLC to operate the workcell.

As to the differences between the prior art and the claims at issue, Banks merely discloses a handbook of simulation in which an entity can be dynamic in that it "moves" through the system, verification of an operational model, and validation of the conceptual model being an accurate representation of the real system. Banks lacks constructing a flowchart that describes interaction of an operator in a workcell using a computer wherein such interaction comprises sequential operations and asynchronous operations, the asynchronous operations being not time dependent, and modeling the operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on predefined conditions described in the flowchart. Banks also lacks testing the control model by a PLC logical verification system on the computer as to whether PLC logic for the workcell is correct and loading the PLC logic in the PLC controlling the workcell if the PLC logic for the workcell is correct. In Banks, there is no logical modeling of operator interaction with a programmable logic controller logical verification system and there are no asynchronous operations of the operator. Also in Banks, there is no modeling of an operator as an input to a programmable logic controller (PLC). Further, Banks is not used to debug PLC logic.

Schruben merely discloses that an event graph can be used to develop alternative event-oriented representations of a system in which several candidate model structures can be considered for possible implementation as discrete-event simulations using an event-scheduling approach. Schruben lacks constructing a flowchart that describes interaction of an operator in a workcell using a computer wherein such interaction comprises sequential operations and asynchronous operations, the asynchronous operations being not time dependent. In Schruben, there are discrete event simulations, which are time based, and cannot account for asynchronous

operations. For example, the (random) time required to repair a jammed machine is modeled as a discrete or time based event because it is denoted by "t" and, therefore, cannot be an asynchronous operation. The passages from Schruben cited by the Examiner do not constitute constructing a flowchart that describes interaction of an operator in a workcell using a computer wherein such interaction comprises sequential operations and asynchronous operations, the asynchronous operations being not time dependent. In Schruben, an operator may be responsible for loading and unloading parts that are processed by a machine as well as freeing a jammed machine, but this operator interaction is <u>not</u> modeled by a flowchart. Further, there is no modeling of an operator as an input to a programmable logic controller (PLC).

As to the level of ordinary skill in the pertinent art, in Banks, an entity can be dynamic in that it "moves" through the system. In Schruben, discrete event simulations are used for time dependent events and do not allow for asynchronous operations, which are <u>not</u> time dependent. Further, neither reference allows for modeling of an operator as an input to a programmable logic controller (PLC). As such, there is absolutely <u>no teaching</u> of a level of skill in the programmable logic controller art that a method of logical modeling operator interaction with a programmable logic controller logical verification system includes the steps of constructing a flowchart that describes interaction of an operator in a workcell using a computer wherein such interaction comprises sequential operations and asynchronous operations, the asynchronous operations being not time dependent, modeling the operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on predefined conditions described in the flowchart, and testing the control model by a PLC logical verification system on the computer as to whether PLC logic for the workcell is correct. The Examiner may not, because he doubts that the invention is patentable,

resort to speculation, unfounded assumptions or hindsight reconstruction to supply deficiencies in the factual basis. See In re Warner, 379 F. 2d 1011, 154 U.S.P.Q. 173 (C.C.P.A. 1967).

Even if the references could be combined, they do not teach a level of skill in the art of programmable logic controller of logical modeling operator interaction with a programmable logic controller logical verification system includes the steps of modeling the operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on predefined conditions described in the flowchart. Applicants are not attacking the references individually, but are clearly pointing out that each reference is deficient and, if combined (although Applicants maintain that they are not combinable), the combination is deficient. The present invention sets forth a unique and non-obvious combination of a method for logical modeling of operator interaction with a programmable logic controller logical verification system that allows a user to verify that the PLC code being planned will work as intended, prior to physically building the tools/manufacturing line and locating equipment. Unlike the prior art, the focus of the present invention is on the logical representation of the operator and not the visual or spatial representations of the operator. Contrary to the Examiner, this is reflected in the claim language because it recites the step of modeling the operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on predefined conditions described in the flowchart. Further, the additional claim language of "the asynchronous operations being not time dependent" was stated by the Examiner in the Advisory Action to overcome the Schruben's reference description of randomly occurring events by explicitly excluding any "time dependence" for "asynchronous operations".

In addition, the Examiner has adduced no factual basis to support his/her position that it would have been obvious to one of ordinary skill in the art to include the operator and

operator's interaction with the PLC-controlled machinery in the PLC logical verification system taught by Banks and that the modification could comprise a "simulated operator" pushing a START or RESTART button on a PLC-controlled machine after "freeing a jammed machine". Thus, the Examiner's stated conclusion of obviousness is based on speculation, unfounded assumptions or hindsight reconstruction to supply deficiencies in the factual basis.

The references, if combinable, fail to teach or suggest the combination of a method of logical modeling operator interaction with a programmable logic controller logical verification system including the steps of constructing a flowchart that describes interaction of an operator in a workcell using a computer wherein such interaction comprises sequential operations and asynchronous operations, the asynchronous operations being not time dependent, modeling the operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on predefined conditions described in the flowchart, testing the control model by a PLC logical verification system on the computer as to whether PLC logic for the workcell is correct, loading the PLC logic in the PLC controlling the workcell if the PLC logic for the workcell is correct, and using the PLC logic by the PLC to operate the workcell as claimed by Applicants. Thus, the Examiner has failed to establish a case of prima facie obviousness.

Against this background, it is submitted that the present invention of claims 1 through 8 is not obvious in view of a proposed combination of Banks and Schruben. The references cannot be combined to render obvious the combination of the method of logical modeling operator interaction with a programmable logic controller logical verification system of claims 1 through 8. Therefore, it is respectfully submitted that claims 1 through 8 are not obvious and are allowable over the rejection under 35 U.S.C. § 103.

Claims 9 through 14

As to claim 9, claim 9 claims the present invention as a method of logical modeling operator interaction with a programmable logic controller logic verification system. The method includes the steps of constructing a flowchart that describes a series of commands for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent. The method also includes the steps of modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart. The method includes the steps of testing the control model by starting a timer and executing the commands by a PLC logical verification system on the computer to test whether PLC logic for the workcell is correct. The method further includes the steps of loading the PLC logic in the PLC controlling the workcell if the PLC logic is correct and using the PLC logic by the PLC to operate the workcell.

As to the differences between the prior art and the claims at issue, Banks merely discloses a handbook of simulation in which an entity can be dynamic in that it "moves" through the system, verification of an operational model, and validation of the conceptual model being an accurate representation of the real system. Banks lacks constructing a flowchart that describes a series of commands for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent, and modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart. Banks also lacks testing the control model by a starting a timer and executing the commands by a PLC logical verification system on the

computer as to test whether PLC logic for the workcell is correct and loading the PLC logic in the PLC controlling the workcell if the PLC logic for the workcell is correct. In Banks, there is no logical modeling of operator interaction with a programmable logic controller logical verification system and there are no asynchronous operations of the operator. Also in Banks, there is no modeling of a human operator as an input to a programmable logic controller (PLC). Further, Banks is not used to debug PLC logic.

Schruben merely discloses that an event graph can be used to develop alternative event-oriented representations of a system in which several candidate model structures can be considered for possible implementation as discrete-event simulations using an event-scheduling approach. Schruben lacks constructing a flowchart that describes a series of commands for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent. In Schruben, there are discrete event simulations, which are time based, and cannot account for asynchronous operations. For example, the (random) time required to repair a jammed machine is modeled as a discrete or time based event because it is denoted by "t" and, therefore, cannot be an asynchronous operation. The passages from Schruben cited by the Examiner do not constitute constructing a flowchart that describes interaction of a human operator in a workcell using a computer wherein such interaction comprises sequential operations and asynchronous operations. In Schruben, an operator may be responsible for loading and unloading parts that are processed by a machine as well as freeing a jammed machine, but this operator interaction is not modeled by a flowchart. Further, there is no modeling of a human operator as an input to a programmable logic controller (PLC).

As to the level of ordinary skill in the pertinent art, there is absolutely no teaching of a level of skill in the programmable logic controller art that a method of logical modeling

operator interaction with a programmable logic controller logical verification system includes the steps of constructing a flowchart that describes a series of commands for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent, modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart, and testing the control model by starting a timer and executing the commands by a PLC logical verification system on the computer as to whether PLC logic for the workcell is correct. The Examiner may not, because he doubts that the invention is patentable, resort to speculation, unfounded assumptions or hindsight reconstruction to supply deficiencies in the factual basis. See In re Warner, 379 F. 2d 1011, 154 U.S.P.Q. 173 (CCPA 1967).

The present invention sets forth a unique and non-obvious combination of a method for logical modeling of operator interaction with a programmable logic controller logical verification system that allows a user to verify that the PLC code being planned will work as intended, prior to physically building the tools/manufacturing line and locating equipment. Unlike the prior art, the focus of the present invention is on the logical representation of the operator and not the visual or spatial representations of the operator. Contrary to the Examiner, this is reflected in the claim language because it recites the step of modeling the operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on predefined conditions described in the flowchart. Further, the additional claim language of "the asynchronous operations being not time dependent" was stated by the Examiner in the Advisory Action to overcome the Schruben's reference description of randomly occurring events by explicitly excluding any "time dependence" for "asynchronous operations".

In addition, the Examiner has adduced no factual basis to support his/her position that it would have been obvious to one of ordinary skill in the art to include the operator and operator's interaction with the PLC-controlled machinery in the PLC logical verification system taught by Banks and that the modification could comprise a "simulated operator" pushing a START or RESTART button on a PLC-controlled machine after "freeing a jammed machine". Thus, the Examiner's stated conclusion of obviousness is based on speculation, unfounded assumptions or hindsight reconstruction to supply deficiencies in the factual basis.

The references, if combinable, fail to teach or suggest the combination of a method of logical modeling operator interaction with a programmable logic controller logic verification system including the steps of constructing a flowchart that describes a series of commands for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent, modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart, testing the control model by starting a timer and executing the commands by a PLC logical verification system on the computer to test whether PLC logic for the workcell is correct, and loading the PLC logic in the PLC controlling the workcell if the PLC logic is correct and using the PLC logic by the PLC to operate the workcell as claimed by Applicants.

Obviousness under § 103 is a legal conclusion based on factual evidence (In re Fine, 837 F.2d 1071, 1073, 5 U.S.P.Q.2d 1596, 1598 (Fed. Cir. 1988), and the subjective opinion of the Examiner as to what is or is not obvious, without evidence in support thereof, does not suffice. Since the Examiner has not provided a sufficient factual basis which is supportive of his

position (see <u>In re Warner</u>, 379 F.2d 1011, 1017, 154 U.S.P.Q. 173, 178 (C.C.P.A. 1967), <u>cert. Denied</u>, 389 U.S. 1057 (1968)), the rejection of claims 9 through 14 is improper.

Against this background, it is submitted that the present invention of <u>claims 9 through 14 is not obvious</u> in view of a proposed combination of Banks and Schruben. The references cannot be combined to render obvious the combination of the method of logical modeling operator interaction with a programmable logic controller logic verification system of claims 9 through 14. Therefore, it is respectfully submitted that <u>claims 9 through 14 are not obvious</u> and are allowable over the rejection under 35 U.S.C. § 103.

<u>Claim 15</u>

As to claim 15, claim 15 claims a method of logical modeling operator interaction with a programmable logic controller logic verification system. The method includes the steps of constructing a flowchart of a series of commands having at least one resource with at least one capability for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent. The method also includes the steps of modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart. The method includes the steps of initializing the operator interaction, idling the operator, testing the control model by starting a timer, executing the commands by a PLC logical verification system on the computer, and determining whether the commands are completed within a predetermined time to test whether PLC logic for the workcell is correct. The method further includes the steps of loading

the PLC logic in the PLC controlling the workcell if the PLC logic is correct and using the PLC logic by the PLC to operate the workcell.

As to the differences between the prior art and the claims at issue, Banks merely discloses a handbook of simulation in which an entity can be dynamic in that it "moves" through the system, verification of an operational model, and validation of the conceptual model being an accurate representation of the real system. Banks lacks constructing a flowchart of a series of commands having at least one resource with at least one capability for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent, and modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart. Banks also lacks testing the control model by starting a timer, executing the commands by a PLC logical verification system on the computer, and determining whether the commands are completed within a predetermined time to test whether PLC logic for the workcell is correct. In Banks, there is no logical modeling of operator interaction with a programmable logic controller logical verification system and there are no asynchronous operations of the operator. Also in Banks, there is no modeling of a human operator as an input to a programmable logic controller (PLC). Further, Banks is not used to debug PLC logic.

Schruben merely discloses that an event graph can be used to develop alternative event-oriented representations of a system in which several candidate model structures can be considered for possible implementation as discrete-event simulations using an event-scheduling approach. Schruben lacks constructing a flowchart of a series of commands having at least one resource with at least one capability for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the

asynchronous operations being not time dependent. In Schruben, there are discrete event simulations, which are time based, and cannot account for asynchronous operations. For example, the (random) time required to repair a jammed machine is modeled as a discrete or time based event because it is denoted by "t" and, therefore, cannot be an asynchronous operation. The passages from Schruben cited by the Examiner do not constitute constructing a flowchart that describes interaction of an operator in a workcell using a computer wherein such interaction comprises sequential operations and asynchronous operations. In Schruben, an operator may be responsible for loading and unloading parts that are processed by a machine as well as freeing a jammed machine, but this operator interaction is <u>not</u> modeled by a flowchart. Further, there is no modeling of a human operator as an input to a programmable logic controller (PLC).

As to the level of ordinary skill in the pertinent art, there is absolutely <u>no teaching</u> of a level of skill in the programmable logic controller art that a method of logical modeling operator interaction with a programmable logic controller logical verification system includes the steps of constructing a flowchart of a series of commands having at least one resource with at least one capability for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent, modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart, and testing the control model by starting a timer, executing the commands by a PLC logical verification system on the computer, and determining whether the commands are completed within a predetermined time to test whether PLC logic for the workcell is correct. The Examiner may not, because he doubts that the invention is patentable, resort to speculation, unfounded assumptions or hindsight reconstruction

to supply deficiencies in the factual basis. See In re Warner, 379 F. 2d 1011, 154 U.S.P.Q. 173 (CCPA 1967).

Even if the references could be combined, they do not teach a level of skill in the art of programmable logic controller of logical modeling operator interaction with a programmable logic controller logical verification system includes the steps of modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart. Applicants are not attacking the references individually, but are clearly pointing out that each reference is deficient and, if combined (although Applicants maintain that they are not combinable), the combination is deficient. The present invention sets forth a unique and non-obvious combination of a method for logical modeling of operator interaction with a programmable logic controller logical verification system that allows a user to verify that the PLC code being planned will work as intended, prior to physically building the tools/manufacturing line and locating equipment. Unlike the prior art, the focus of the present invention is on the logical representation of the operator and not the visual or spatial representations of the operator. Contrary to the Examiner, this is reflected in the claim language because it recites the step of modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on predefined conditions described in the flowchart. Further, the additional claim language of "the asynchronous operations being not time dependent" was stated by the Examiner in the Advisory Action to overcome the Schruben's reference description of randomly occurring events by explicitly excluding any "time dependence" for "asynchronous operations".

In addition, the Examiner has adduced no factual basis to support his/her position that it would have been obvious to one of ordinary skill in the art to include the operator and

operator's interaction with the PLC-controlled machinery in the PLC logical verification system taught by Banks and that the modification could comprise a "simulated operator" pushing a START or RESTART button on a PLC-controlled machine after "freeing a jammed machine". Thus, the Examiner's stated conclusion of obviousness is based on speculation, unfounded assumptions or hindsight reconstruction to supply deficiencies in the factual basis.

The references, if combinable, fail to teach or suggest the combination of a method of logical modeling operator interaction with a programmable logic controller logic verification system including the steps of constructing a flowchart that describes a series of commands for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent, modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart, testing the control model by starting a timer and executing the commands by a PLC logical verification system on the computer to test whether PLC logic for the workcell is correct, and loading the PLC logic in the PLC controlling the workcell if the PLC logic is correct and using the PLC logic by the PLC to operate the workcell as claimed by Applicants.

Obviousness under § 103 is a legal conclusion based on factual evidence (In re Fine, 837 F.2d 1071, 1073, 5 U.S.P.Q.2d 1596, 1598 (Fed. Cir. 1988), and the subjective opinion of the Examiner as to what is or is not obvious, without evidence in support thereof, does not suffice. Since the Examiner has not provided a sufficient factual basis which is supportive of his position (see In re Warner, 379 F.2d 1011, 1017, 154 U.S.P.Q. 173, 178 (C.C.P.A. 1967), cert. Denied, 389 U.S. 1057 (1968)), the rejection of claims 15 through 20 is improper.

Against this background, it is submitted that the present invention of claim 15 is not obvious in view of a proposed combination of Banks and Schruben. The references cannot be combined to render obvious the combination of the method of logical modeling operator interaction with a programmable logic controller logic verification system of claim 15. Therefore, it is respectfully submitted that claim 15 is not obvious and is allowable over the rejection under 35 U.S.C. § 103.

CONCLUSION

In conclusion, it is respectfully submitted that the rejection of claims 1 through 15 is improper and should be reversed.

Respectfully submitted,

By: Daniel H. Bliss

Daniel H. Bliss
Registration No. 32,398

BLISS McGLYNN, P.C.
2075 West Big Beaver Road
Suite 600
Troy, Michigan 48084-3443
(248) 649-6090

Dated: July 16, 2007

Attorney Docket No.: 0693.00257
Ford Disclosure No.: 200-0665

CLAIMS APPENDIX

The claims on appeal are as follows:

1. A method of logical modeling operator interaction with a programmable logic controller logical verification system, said method comprising the steps of:

constructing a flowchart that describes interaction of an operator in a workcell using a computer wherein such interaction comprises sequential operations and asynchronous operations, the asynchronous operations being not time dependent;

modeling the operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on predefined conditions described in the flowchart;

testing the control model by a PLC logical verification system on the computer as to whether PLC logic for the workcell is correct; and

loading the PLC logic in the PLC controlling the workcell if the PLC logic for the workcell is correct and using the PLC logic by the PLC to operate the workcell.

2. A method as set forth in claim 1 wherein the step of testing comprises starting a timer and determining whether the operator interaction of the flowchart is completed within a predetermined time.

3. A method as set forth in claim 2 wherein the step of testing includes initializing the operator interaction of the flowchart prior to starting the timer.

4. A method as set forth in claim 3 wherein said step of testing includes idling the operator prior to starting the timer.

5. A method as set forth in claim 1 wherein said step of constructing comprises constructing a series of commands for the operator using the computer.

6. A method as set forth in claim 5 wherein the commands have at least one resource.

7. A method as set forth in claim 6 wherein the at least one resource has at least one capability.

8. A method as set forth in claim 1 wherein the step of testing includes executing the commands when a timer is started.

9. A method of logical modeling operator interaction with a programmable logic controller logic verification system, said method comprising the steps of:

constructing a flowchart that describes a series of commands for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent;

modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart;

testing the control model by starting a timer and executing the commands by a PLC logical verification system on the computer to test whether PLC logic for the workcell is correct; and

loading the PLC logic in the PLC controlling the workcell if the PLC logic is correct and using the PLC logic by the PLC to operate the workcell.

10. A method as set forth in claim 9 wherein the step of testing includes determining whether the commands of the flowchart are completed within a predetermined time.

11. A method as set forth in claim 10 wherein the step of testing includes initializing the operator interaction of the flowchart prior to starting the timer.

12. A method as set forth in claim 11 wherein said step of testing includes idling the operator prior to starting the timer.

13. A method as set forth in claim 9 wherein said step of constructing comprises constructing commands having at least one resource.

14. A method as set forth in claim 13 wherein the at least one resource has at least one capability.

15. A method of logical modeling operator interaction with a programmable logic controller logic verification system, said method comprising the steps of:

constructing a flowchart of a series of commands having at least one resource with at least one capability for a human operator in a workcell using a computer wherein such commands comprise sequential operations and asynchronous operations, the asynchronous operations being not time dependent;

modeling the human operator as an input to a programmable logic controller (PLC) by writing a control model of the operator interaction in the workcell based on the commands in the flowchart;

initializing the operator interaction and idling the operator;

testing the control model by starting a timer, executing the commands by a PLC logical verification system on the computer, and determining whether the commands are completed within a predetermined time to test whether PLC logic for the workcell is correct; and

loading the PLC logic in the PLC controlling the workcell if the PLC logic is correct and using the PLC logic by the PLC to operate the workcell.

RELATED PROCEEDINGS APPENDIX

None